# CORPORATE INFORMATION MANAGEMENT (CIM)

# REPORT ON THE SIMULATION OF PROCESSES

**Prepared by**

**ANSER**

**15 April 1993**



**Director of Defense Information**
**Office of the Assistant Secretary of Defense**
**(Command, Control, Communications, and Intelligence)**
**Washington, DC 20301-3040**

15 April 93

Corporate Information Management (CIM)
Report on the Simulation of Processes
Version 2.0, 22 Jun3 1993

John Tieso

Office of the Deputy Assistant Secretary of Defense
(Information Systems)

Distribution A
Approved for Public Release;
Distribution is Unlimited

This report discusses three pilot projects simulating command and control processes. The projects chosen were: Message Processing Center (MPC), Integrated Satellite Control System (ISCS), Theater Missile Defense (TMD). All three projects used the commercial off-the-shelf (COTS) simulation tool, Design/CPN. Design/CPN investigates color petri nets (CPNs) as a mathematical or algorithmic underpinning for analyzing complex information systems and validates claims that Design/CPN could be viewed as an extension of IDEF0 activity models. Other tools and algorithms were selectively considered for comparisons.

79

CIM Collection

Unclassified Unclassified UL

The Report [illegible]
that this [illegible]
Instructions [illegible]
*optical scanning* [illegible]

Block [illegible]

Block [illegible]

Block [illegible]

Block 5. [illegible]
and [illegible]
element [illegible]
number [illegible]

C [illegible]
G [illegible]
PE [illegible]

Block [illegible]

CORPORATE INFORMATION MANAGEMENT (CIM)

REPORT ON THE SIMULATION OF PROCESSES

Prepared by

ANSER

15 April 1993




Director of Defense Information
Office of the Assistant Secretary of Defense
(Command, Control, Communications, and Intelligence)
Washington, DC 20301-3040

## EXECUTIVE SUMMARY

### SUMMARY OF PROJECTS

This report discusses three pilot projects simulating command and control processes. The projects chosen were

| Project | User Client |
|---|---|
| *Message Processing Center (MPC)* | *Joint Interoperability & Engineering Organization (JIEO)* |
| *Integrated Satellite Control System (ISCS)* | *U.S. Space Command (USSPACECOM)* |
| *Theater Missile Defense (TMD)* | *Strategic Defense Initiative Office (SDIO)* |

All three projects used the commercial off-the-shelf (COTS) simulation tool, Design/CPN. Design/CPN investigates color petri nets (CPNs) as a mathematical or algorithmic underpinning for analyzing complex information systems and validates claims that Design/CPN could be viewed as an extension of IDEF0 activity models. Other tools and algorithms were selectively considered for comparisons.

### The MPC Project

The MPC provided a simple but poignant illustration of the power of petri net modeling in the context of process improvement. The project model was a MPC of a large headquarters during a major command post exercise in NATO. The problem considered involved an unacceptably large backlog of messages accumulating during the exercise. An issue was raised on how this problem could best be solved. Two alternative solutions were available: either increase manpower or apply technology. Without simulation, neither option could be adequately evaluated. The simulation, however, demonstrated that given either of the options, increasing manpower yielded the most productive solution. The MPC project was important in understanding how to apply simulation to a process for improvement and yielded a result which is a simple example for those who want to learn about process simulation.

### The ISCS Project

The ISCS project was more complicated than the MPC in that it involved both physical and informational phenomena and a greater number of processes. ISCS is a defense program to integrate all of the Service satellite control systems into one interoperable system. The physical problem involves having to communicate from one of many remote tracking stations distributed about the globe where each satellite has its own orbit. Orbits are geosynchronous, geosolar, or elliptical rotating and of varying periods. Competing windows of opportunity for communications appear and disappear daily. Information issues are generated by messages of different priority, length, and purposes competing for processing times. Generally messages are categorized into those impacting the health or welfare of the satellite platform and those impacting its mission (communications, intelligence, environmental, etc.). Because of the size of the entire

ISCS program, only a subset the Defense Meteorological Satellite Program (DMSP), was modeled. A scenario was developed that focused on CINC requirements for DMSP support. The scenario was chosen to address issues from Desert Storm involving the tasking of satellite support for the CINC. This simulation provides an excellent example of how fixing a process can be more cost-effective than buying expensive technology.

**The TMD Project**

The TMD project generally was constructed to address interoperability issues between the Services. The project selected by the SDIO, called the Counterforce Options problem, was constructed around a scenario that involved choosing between competing Service capabilities for an attack on theater missile launcher. Examples of competing capabilities could include an Army Tactical Missile System (ATACMS) and Air Force Fighter (F-16) or naval gunfire. The purpose of the simulation was to develop a capability to develop or assess rules of engagement. An interesting aspect of the Counterforce Options simulation is that it was developed from an existing IDEF0 activity model that had been prepared to support studies by the Phase One Engineering Team (POET) for the SDIO. ANSER used the automatic programming facility of Design/IDEF to convert the IDEF0 activity model to a CPN model, validating the value of this feature. This project also illustrates how simulation can be used to develop and assess doctrine and associated procedures.

**FINDINGS**

All of the projects were successful in meeting the established objectives. Collectively they produced the following findings:

1. There is a need to develop DOD policy and procedures on the application of process simulation.

2. Relationships between process simulation and other functional process improvement methodologies were determined to be

- Simulation provides a temporal perspective to process modeling
- Simulation can quantify process costs and benefits and can support the development of a functional economic analysis.

3. Process simulation provides a means of analyzing and assessing temporal issues in processes. Simulation can reveal

- Resource flow bottlenecks
- Idle processes
- Global value of local changes.

4. The simulation projects accredited Design/CPN as an effective tool in analyzing processes. Some interesting features of this tool included

- Object-oriented design features, minimizing programming and facilitating the construction of a simulation in a logical way

- The graphical interface, facilitating setup and analysis

- Powerful analytical capability. Petri nets were designed to address complexity inherent in information management. For instance, they deal with synchronicity, parallelism, and conflict for resources, which are all nonlinear concepts that are difficult for other simulation techniques

- The automatic program features of Design/IDEF that convert IDEF0 diagrams to CPN diagrams for Design/CPN.

5. Data for process simulations should be collected during activity modeling for two reasons:

- It forces an activity modeler to better understand the processes in the activity model

- It assures continuity of analysis between activity models and associated simulations.

6. Generic data required for simulations include

- Procedures for defining a process cycle, particularly initiation, intention, and termination control of a cycle

- Time required for each cycle

- Number of cycles required of a child process for each cycle of its parent process

- Quantity and value (cost and utility) of resources consumed by a process.

## RECOMMENDATIONS

**1. The Director of Defense Information should develop and publish procedures for applying process simulation as a CIM process improvement methodology. Procedures should focus on temporal issues and should include itemizing data collection during activity or data modeling that will facilitate process simulation.**

2. The Office of the Secretary of Defense should conduct independent verification, validation, and accreditation to the extent feasible of process simulation tools (e.g., Design/CPN, MODELER, ITHINK, SIMPROCESS, PACE, and others as appropriate).

# PREFACE

This project required ANSER to model three command and control systems as specified by the government. The objective was to demonstrate the utility of simulation to command and control and to make general observations about the application of simulation to processes as a CIM improvement methodology.

Three projects were chosen by the government in discussion with ANSER. Simulations included a message processing center (MAC) activity supporting a major NATO exercise, WINTEX/ CIMEX; a component of the Integrated Satellite Control System (ISCS); and Theater Missile Defense (TMD). The projects were selected based on their potential to provide the broadest observations that could be made within the level of resources provided.

Work on the projects included the efforts of ANSER employees Mr. Gary Coe, Mr. Don Flint, Mr. Al Mielus, Ms. Lisa Shogren, Ms. Melissa Young, and Ms. Michele Arnold.

Electronic files of the simulations have been provided to the government through the Contractor Officer Technical Representative (COTR), Major Michael McCullough.

This task is sponsored by the Director of Defense Information (DDI) and is administered under the direction of JIEO of the Defense Information Systems Agency (DISA) through a contract with LOGICON, DAAB07-91-B519. ANSER is a subcontractor to LOGICON.

## REPORT ON PROCESS SIMULATION

## CONTENTS


| | | Page |
|---|---|---|
| 1.0 | Introduction | 1 |
| | 1.1 Tasking | 1 |
| | 1.2 Subprojects | 1 |
| | 1.3 Analysis of Approach | 1 |
| | 1.4 Data Analysis | 3 |
| | 1.5 Tools | 3 |
| | 1.6 Caveat | 3 |
| 2.0 | Message Processing Center | 4 |
| | 2.1 Background | 4 |
| | 2.2 Objective | 4 |
| | 2.3 Simulation Methods, Analysis, and Results | 7 |
| | 2.4 Observations | 9 |
| 3.0 | Integrated Satellite Control System | 10 |
| | 3.1 Background | 10 |
| | 3.2 Objective | 11 |
| | 3.3 Simulation Methods, Analysis, and Results | 12 |
| | 3.4 Observations | 15 |
| 4.0 | Theater Missile Defense | 17 |
| | 4.1 Background | 17 |
| | 4.2 Objective | 17 |
| | 4.3 Simulation Methods, Analysis, and Results | 17 |
| | 4.4 Observations | 26 |
| 5.0 | Discussion | 27 |
| | 5.1 General | 27 |
| | 5.2 Policy Issues | 27 |
| | 5.3 Functional Process Improvement | 27 |
| | 5.4 Analysis Issues | 27 |
| | 5.5 Data Collection | 28 |
| | 5.6 Simulation Tools | 28 |

**CONTENTS-Continued**


| | | Page |
|---|---|---|
| 6.0 | Findings | 30 |
| 7.0 | Recommendations | 32 |
| Appendix A--Diagram of Message Processing Center | | A-1 |
| Appendix B--Diagrams of Integrated Satellite Control System | | B-1 |
| Appendix C--Diagrams of Theater Missile Defense | | C-1 |
| Appendix D--Bibliography | | D-1 |
| Appendix E--Distribution List | | E-1 |

CONTENTS--Continued

## FIGURES


| Figure | | Page |
|---|---|---|
| 2.1 | AO: Message Processing Center | 4 |
| 2.2 | AO: Process and Transmit Messages | 5 |
| 2.3 | Message Processing Center Simulation Diagram | 6 |
| 2.4 | Message Processing Center, Simulation Results by Case and Message Priority | 8 |
| 3.1 | DMSP System Elements | 11 |
| 3.2 | Top-Level Model Defense Meteorological Satellite Program | 12 |
| 3.3 | Decomposition of Process Task | 13 |
| 3.4 | Range of Results for Differing Constellations | 14 |
| 4.1 | A2 Node for Both Scenarios | 18 |
| 4.2 | A22 Node for Scenario "A" | 19 |
| 4.3 | IDEF0 Page | 20 |
| 4.4 | CPN Page | 21 |
| 4.5 | CPN Statistical Output | 22 |
| 4.6 | Weapon Assignments | 23 |
| 4.7 | Success Ratios | 23 |
| 4.8 | Average Time | 24 |
| 4.9 | Restrike Results | 24 |

# 1.0 INTRODUCTION

## 1.1 TASKING

ANSER was tasked by the Director of Defense Information (DDI) to produce three process simulations of command and control systems as determined by the government. The work was performed as a subcontract (No. 895/ANS) to LOGICON under government contract DAAB07-91-D-B519 dated 3 December 1991.

## 1.2 SUBPROJECTS

This report discusses three pilot projects to simulate command and control processes. The projects chosen were

| *Project* | *User Client* |
|---|---|
| *Message Processing Center (MPC)* | *Joint Interoperability & Engineering Organization (JIEO)* |
| *Integrated Satellite Control System (ISCS)* | *U.S. Space Command (USSPACECOM)* |
| *Theater Missile Defense (TMD)* | *Strategic Defense Initiative Office (SDIO)* |

All three projects used the commercial off-the-shelf (COTS) simulation tool Design/CPN. Design/CPN investigated of color petri nets (CPNs) as a mathematical or algorithmic underpinning for analyzing complex information systems and claims that Design/CPN could be viewed as an extension of IDEF0 activity models. For comparison, other tools and algorithms were selectively considered.

## 1.3 ANALYSIS OF APPROACH

### 1.3.1 Problem Selection and Definition

Command, control, and communications interoperability, systems dynamics, and data availability were key aspects of subjects selected for simulation.

The message processing center problem met the requirement of being an easily understood command and control system having apparently simple (but actually complex) dynamics with available data. Data were provided by an Air Force officer who served within the MPC during a major NATO exercise, WINTEX/CIMIC.

The ISCS problem was selected by the J5, USSPACECOM and endorsed by the USSPACECOM staff as a command and control interoperability problem needing simulation. The complete ISCS was determined to be too complex for the low level of effort provided, so a subsystem was selected by USSPACECOM for analysis. The subsystem chosen was the Defense Satellite Meteorological Program (DMSP). It was chosen because it lacked immediate political interest, data were available, and a solution to the DMSP problem was analytically interesting and could be easily extended to other ISCS subsystems. A scenario was developed in coordination

with the government involving the tasking by a warfighting CINC of USSPACECOM for satellite support. The problem structure evolved from lessons learned during Desert Storm on satellite tasking by a CINC.

The TMD problem was selected jointly by JIEO and SDIO. JIEO was concerned about whether interoperability of TMD was being adequately addressed by the SDIO and Services. SDIO was concerned about the adequacy of its analytical capability to address tough issues such as how rules of engagement are developed for emerging TMD concepts. The Counterforce Options problem was specifically chosen for its interoperability aspects because of the preliminary work done by the Phase One Engineering Team (POET) assuring data availability and because of its complexity as a problem that would test the character of the simulation tools and approach to be used.

Activity models were developed for each of the problems to be simulated and used to support simulation development. It was determined after the TMD problem was selected that IDEF0 activity models were already in existence for the Counterforce Options problem. This was a fortuitous event, adding to the credibility of the project.

### 1.3.2 Measures and Data Collection

Measures and data collection routines were determined to support postsimulation analysis. Data analysis depends on the purpose of a simulation effort. Nonetheless, the design of such analysis can be guided by some general principles of system theory. Basic measures of performance for information systems include availability, utilization, throughput, response time, workload, and system balance. These measures can apply to the whole system, segments of the system, subsystems, and components of subsystems. Definitions of these measures are shown below:

- **Availability** is the ratio of the number of times the system, segment, subsystem, or component is ready (to be used) to the total number of times it is needed.
- **Utilization** is the ratio of time in use to the time available for use.
- **Throughput** is the level of work done (e.g., the number of things processed or produced over a period of time).
- **Response time** is the elapsed time from the end of an input to the start of a response to threat input.
- **Workload** is the number of inputs or the number of demands per unit of time.
- **System balance** is the distribution of idle, busy, and blocked segments, subsystems, and components during particular periods.

Other measures can be derived from these basic measures. These derived measures include

- **Availability versus throughput**
- **Throughput versus workload**
- **Utilization versus workload**
- **Response time versus workload.**

In the simulation of command and control systems, more specific articulation of measures was required. For instance, response time addressed in terms of windows of opportunity and workload was measured as number of resources used per unit of time.

## 1.4 DATA ANALYSIS

Design/CPN provides a facility for statistic definition and collection. It offers a feature for chart construction during and after simulation. It also provides for exporting data files to other data analysis tools such as spreadsheet or data analysis software. All of these capabilities were used during the simulations developed.

## 1.5 TOOLS

Design/IDEF and Design/CPN were the primary software tools used during the project. They were run on a Macintosh Quadra 900, a high-performance workstation required for simulation.

Other tools considered for comparison purposes were MODELER, an Air Force-developed petri net simulation software similar to Design/CPN; ITHINK, a COTS simulation software using a simulation approach known as systems dynamics; and SIMPROCESS, a COTS simulation software using a SIMSCRIPT programming approach.

## 1.6 CAVEAT

Data used in the projects selected included notional data. Therefore, any conclusions about the situations modeled should be held in reserve. The focus of the projects was to investigate process simulation as an assessment tool. Rules on data collection required the use of unclassified data to keep the projects unclassified or selective notional data to minimize cost in data collection.

## 2.0 MESSAGE PROCESSING CENTER

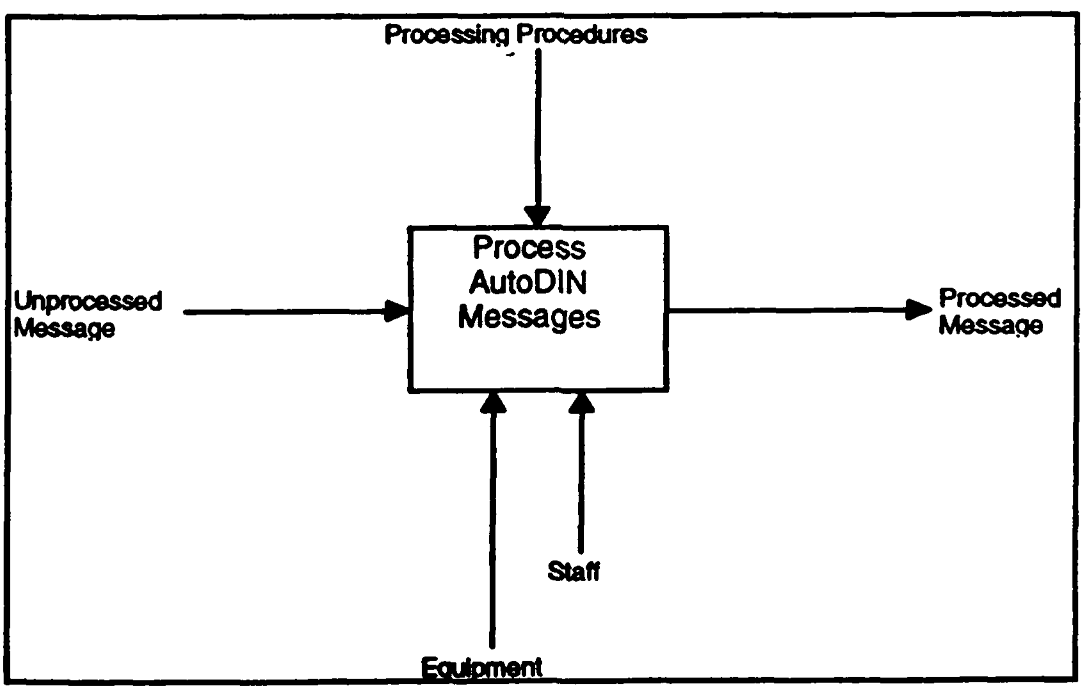
**A Case Example:**
**A Message Processing Center for a Large NATO Headquarters**

### 2.1 BACKGROUND

Activity at a message processing center (MPC) during a military exercise was chosen for a case study of process simulation. The MPC selected was on the Automated Digital Information Network (AutoDIN) system supporting a NATO Southern Region War exercise (WINTEX/CIMEX) in Spain. Messages, manually generated by the headquarters, are processed at the MPC, where they are entered electronically into AutoDIN for transmission to other nodes. Appendix A contains a diagram of a MPC.

### 2.2 OBJECTIVE

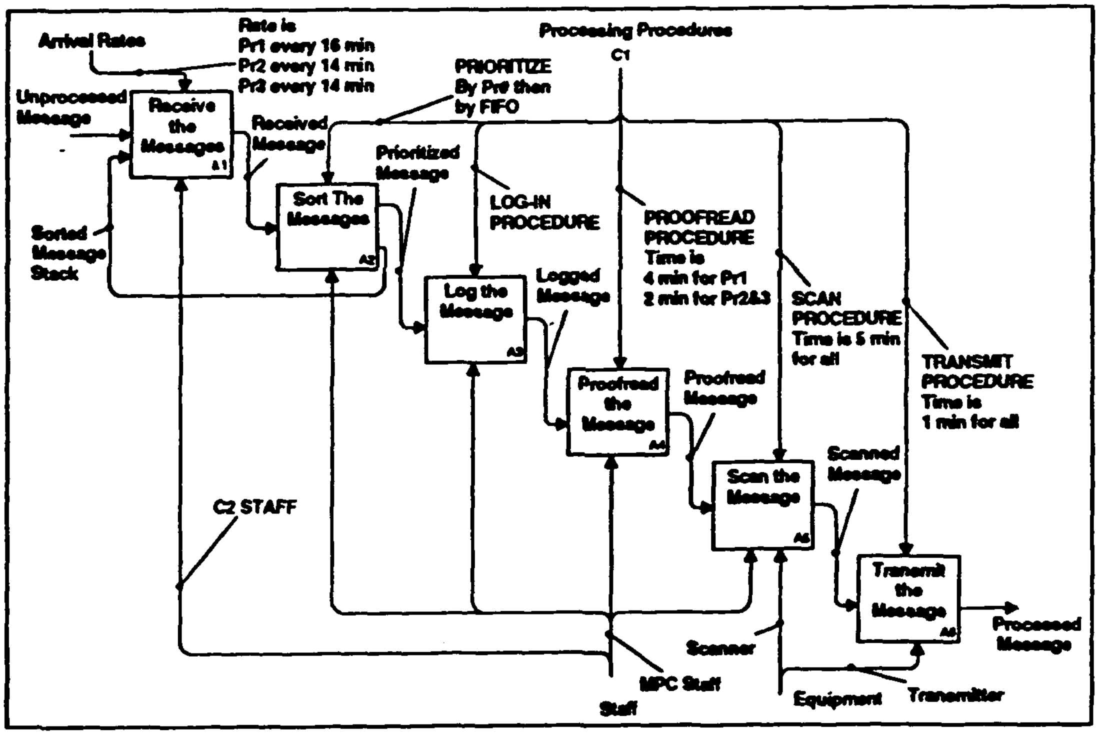
The problem for analysis was a rapidly growing backlog of messages, resulting in unacceptably increasing waiting times for transmission. A desired solution would reduce these waiting times and associated backlogs. This is a common problem of command and control systems that surfaces during exercises and operations where message traffic is significantly higher than during routine activities.




**A0: Message Processing Center**
**Figure 2.1**

### 2.2.1 Details of the MPC Process

Three types of messages arrived at the center: immediate, priority, and status (referred to as priority one, two, and three, respectively). An operator would select a message, giving precedence to priority one messages, review it for errors, and hand feed it into an optical scanner. Once a message was scanned, it was automatically transmitted, and the operator was free to select a new message. Priority one messages arrived approximately every 16 minutes, while priority two and three messages arrived approximately every 14 minutes. Priority one messages took 4 minutes (on the average) to log in and review, while priority two and three messages took 2 minutes. The scanning process took 5 minutes (on the average) to complete regardless of the priority level. Transmission took an average of 1 minute regardless of the priority level. The first two levels of an IDEF0 activity model of the MPC are shown in Figures 2.1 and 2.2 respectively.




**AO: Process and Transmit Messages**
**Figure 2.2**

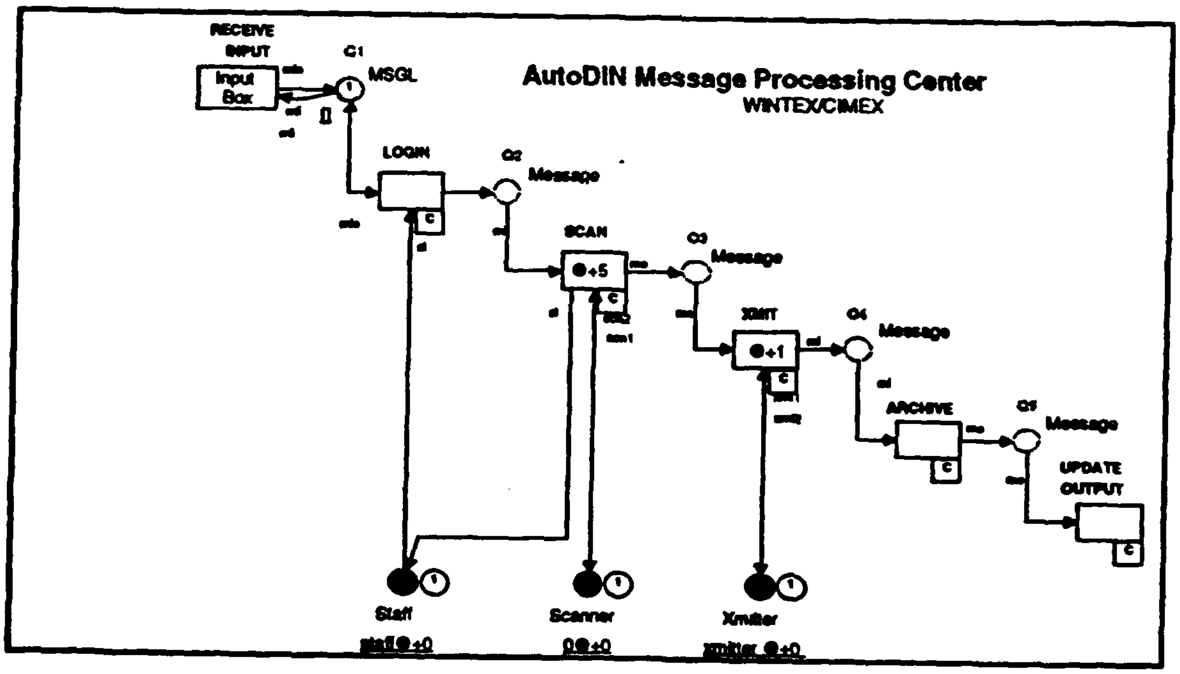
Mechanisms for the A2, A3, and A5 activities shown in Figure 2.2 reveal options for reducing the backlog of messages. Four cases were developed from the following two general options:

- Add manpower to the preprocessing of messages (checking for correct format and priority) and structure the use of manpower.

- Improve the scanning capability by replacing the existing scanner with one that is more responsive or by adding scanners. It was given that funding would be unavailable to upgrade the scanner; however, this case was investigated anyway.

Before simulation was introduced, the MPC supervisor concluded that the message bottleneck was caused by a slow scanner (an average of 5 minutes per message) (see A5 activity). This conclusion was based only on his subjective assessment. However, a member of his team suggested that the backlog would be reduced if manpower were added to the process. A simulation was developed on this hypothesis.

Simulations of the problem above have been developed using four different tools: Design/CPN, MODELER, SIMPROCESS, and ITHINK. Design/CPN and MODELER both use discrete event simulation based on petri net mathematics. ITHINK uses continuous process simulation based on systems dynamics and SIMPROCESS uses the SIMSCRIPT procedural language. All four simulations yielded the same result. A structural representation of the Design/CPN is shown in Figure 2.3.




**Message Processing Center**
**Simulation Diagram**
**Figure 2.3**

## 2.3 SIMULATION METHODS, ANALYSIS, AND RESULTS

### 2.3.1 Methods

Design/CPN was used to model the AutoDIN message processing center. This model is shown in Figure 2.3. It shows *"arcs,"* paths of messages, through nodes of two types. The circle node is called a *"place,"* representing a queue of messages (e.g., buffer) for an activity. The rectangle node is called a *"transition,"* representing the activity. (In activity models, boxes combine the representation of queues and activities.) As intuition would suggest, at least one place precedes every transition, and transition feed only into places. Similarly, places are fed by and feed only into transitions. Petri net *tokens*, which flow through the network, model the resources and associated attributes. Resources modeled included messages and staff. The attributes modeled for messages included

- Priority type
- Arrival time to a transition
- Departure time from a transition
- Other time statistics.

We have omitted some technical description as to exactly how transitions work and how attributes are assigned; however, the simulation is constructed largely in a "Lego-like" fashion where the building blocks are *tokens* (resources), *places* (queues), *transitions* (activities), and *arcs* (relationships between the activities).

In the MPC model, as messages arrive at the center, they are sorted according to priority. This is done with an appropriate code segment, executed each time the transition *Input Box* is enabled. (The term "enabled" means that certain conditions have been satisfied for the activity to begin.) Messages then wait at place *Q1* until the transition *LOGIN* is enabled (there is at least one message at *Q1* and one operator at the place *Staff*). When *LOGIN* is enabled, an output token is created at place *Q2*. The arc inscription adds two or four to the token's time stamp, depending on priority level. The message waits until the transition scan is enabled (there is at least one message at *Q2* and at least one token at the place *Scanner* representing that the scanner is available). When *SCAN* is enabled, a token is created at the *OUTPUT* place with an added time delay of five, representing the 5 minutes needed to scan. Similarly, *XMIT* will be enabled if there is at least one message at its two input places. A time delay of one is added to the output token (1 minute is needed to transmit a message). The message then exits the system. Appropriate code segments were also added to keep the time statistics associated with each message. Output was sent to a separate file to be processed. However, line graphs can be built within Design/CPN.

To add a second operator to the model, all that is needed is to change the initial marking (initial condition) of the place *Staff* to two tokens. The foregoing is presented to provide a flavor of what is involved in simulating a process. Any of the measures described in Section 5 can be addressed in this simulation. What concerns us here is throughput and how to increase it.

### 2.3.2 Simulation Analysis (See Figure 2-4)

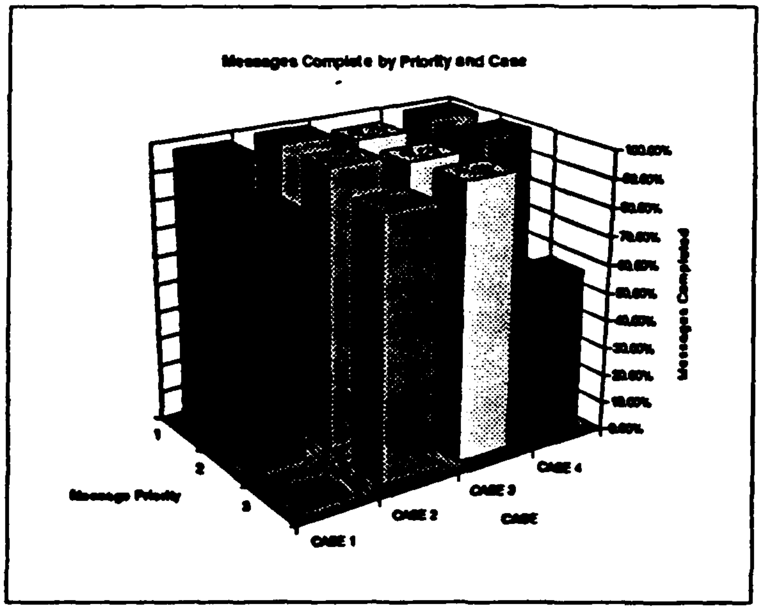
For all four cases 296 messages (24 hours of message flow) were simulated.

- **Case 1 - The baseline: One staff member to log, proofread, and scan messages:** A backlog of 117 (about 40%) messages was generated for this 24-hour simulated period. Only one of the priority one messages and 13 of the priority two messages were in the backlog. Most of the backlog (103 messages) consisted of priority three messages.

- **Case 2 - Two staff members, each logging, proofreading, and scanning messages:** In this case, the backlog was reduced dramatically to a total of nine (no priority one, one priority two, and eight priority three messages).

- **Case 3 - Two staff members, one to log and proofread, and one to scan:** In this case the backlog was reduced to a total of 10 but with an increase in priority one messages to three. Priority two messages were reduced to four, and priority three messages were reduced to three.

- **Case 4 - Baseline modified by reducing scanning time to 3 minutes:** The backlog was reduced to 44 priority three messages.




**Message Processing Center**
**Simulation Results By Case and Message Priority**
**Figure 2.4**

### 2.3.3 Results

While the supervisor was correct in assessing that replacement of the scanner with one having improved performance characteristics would reduce the message backlog (117 to 44 in the case simulated), clearly a better option for reducing the backlog was to add a staff member, preferably as was done in Case 2 above.

## 2.4 OBSERVATIONS

### 2.4.1 General

The MPC project demonstrated the value of simulation to the understanding and qualification of information management issues. While one could argue what intuitively was the best solution to the MPC problem, manpower or technology, the MPC supervisor thought that technology was the solution. Thus the simulation demonstrated counterintuitive results and could do the same potentially for other issues.

### 2.4.2 Implications for Functional Economic Analysis

Two considerations not evaluated in the simulation above are the cost of adding manpower or technology and the value of improved performance. Had these factors been incorporated, we would have the remaining parameters for supporting a functional economic analysis.

Cost can be a relatively easy computation from a salary schedule. However, if cost is seen as an opportunity cost, computation could be more difficult. Opportunity cost is the lost value caused by moving the required manpower from another staff position. Manpower is a scarce resource in military organizations, making the choice of using manpower to be a fundamental economic decision.

Determining value can be difficult. With a MPC, it is important to understand how messages are used. For instance, what activities will be delayed as a result of not receiving a message within a certain time window and what will be the impact of the delay? To determine the answer, it may be appropriate to extend the simulation above to include activities requiring message flow.

In summary, process simulation can provide better cost and value representations in support of functional economic analysis than can be provided by other means.

# 3.0 INTEGRATED SATELLITE CONTROL SYSTEM

**Featuring the**
**Defense Meteorological Satellite Program**

## 3.1 BACKGROUND

The Integrated Satellite Control System (ISCS) was chosen by United States Space Command (USSPACECOM) as a pilot program for simulation in conjunction with other activity modeling efforts. ISCS is a program to integrate the satellite control networks of all the Services into a joint system. Satellite control networks provide spacecraft command and control support; telemetry reception of both payload mission and spacecraft bus data; launch and on-orbit vehicle tracking for DOD satellites; Research, Development, Test and Evaluation (RDT&E) programs; and other assigned space programs. ISCS consists of various information processing segments including Mission Control Centers (MCCs) and Remote Tracking Stations (RTSs). RTSs provide satellite-to-ground command and control interface operations under the direction of an MCC. MCCs include manpower, hardware, and software necessary to receive, process, and transmit data through the ISCS network. Specifically, ANSER was tasked to build, as proof of concept, a model depicting satellite control for a satellite network consisting of one MCC, one RTS, and one satellite.

The Defense Meteorological Satellite Program (DMSP) was chosen for the pilot program because the physics of the DMSP could easily be extended to other programs. It represented realistic issues as observed during Desert Storm, while the program analysis seemed politically benign.

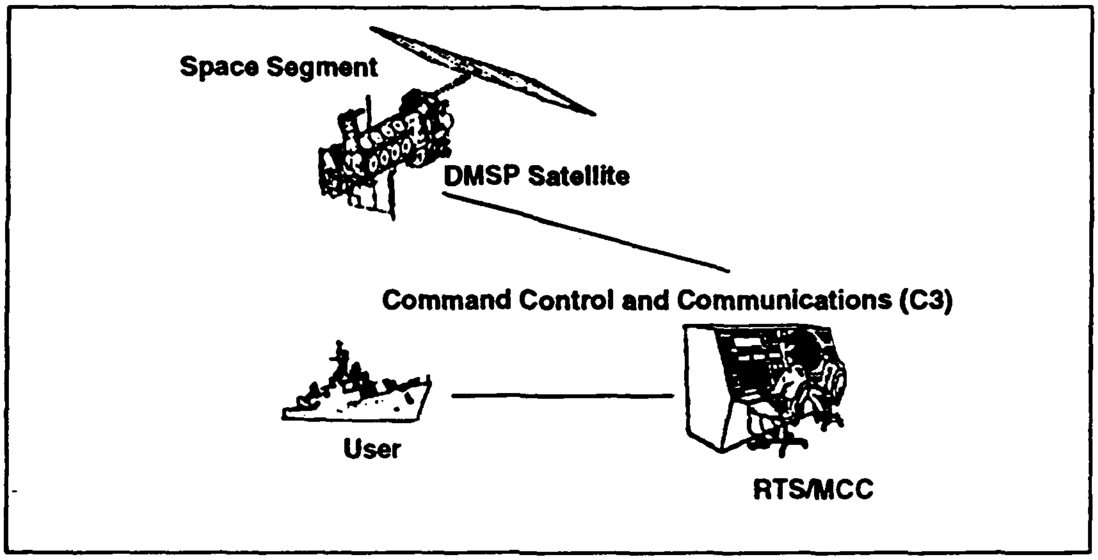
Other reasons the government chose DMSP for this proof-of-concept model included availability of data, ability to build an unclassified model, and ability to demonstrate the modeling capability with a fairly simple system. We also considered possible controversies that could have developed if we tried to model a proposed or existing system over which inter-agency control was in question. DMSP avoided such controversies.

The DMSP provides an enduring and survivable capability to collect and disseminate global visible and infrared cloud data and other specialized meteorological, oceanographic, and solar geophysical data required to support worldwide DOD operations and high-priority programs. It is the primary source of meteorological satellite data for the DOD.

The DMSP consists of three segments as shown in Figure 3.1: Command, Control and Communications (C3), and User. The Space segment consists of the constellation of satellites that perform the actual collection of the data. A minimum of two on-orbit operational satellites is maintained at all times. Additional satellites are launched as needed to offset the degradation of the satellites through time.

The C3 segment provides all of the functions required to maintain state of health of the DMSP satellites and provides communications media to recover the sensor payload data acquired during the satellites' orbit. RTSs in Greenland (Thule Tracking Station) and Hawaii

(Hawaii Tracking Station) provide access to the satellites, which is limited to a maximum of 15 minutes during each 101-minute orbit. During the access time, the RTS must (1) command the satellites in real time, (2) uplink commands to be executed when it is not in direct contact, (3) download and analyze real-time health and welfare telemetry of the satellite, (4) download telemetry data for off-line analysis, and (5) download the sensor payload data.




*DMSP System Elements*
**Figure 3.1**

The User segment of the DMSP is composed of strategic and tactical users. The strategic users are the Air Force Global Weather Center (AFGWC) and the Fleet Numerical Oceanography Center (FNOC). The AFGWC is responsible for processing and distributing strategic meteorological and mission sensor data to users. The FNOC provides the Navy with analyses and forecasts of oceanographic and marine weather parameters, including air and subsurface, at any global location.

The tactical users of the system are the United States Air Force (USAF), United States Marine Corps (USMC), and the United States Navy (USN). Tactical users employ tactical terminal vans or ship-based terminals for direct readout of real-time visible and infrared cloud cover data from the satellites. The tactical terminals do not have the capability to receive and process stored spacecraft payload data, nor are they able to command the satellites, either to affect the orbit or to order sensor data for distant locations.

## 3.2 OBJECTIVE

In order to limit the scope of the model to a reasonable size and yet provide the details of interest, we chose to model from a warfighter's or Joint Task Force (JTF) point of view. The question we were trying to answer was, "How long does it take for the warfighter to receive specialized data in response to a request for any location?" By answering that question we

would be able to show where the backups were occurring, which subtasks were taking the longest to perform, and how efficient the method was of acquiring the data and to answer other performance questions.

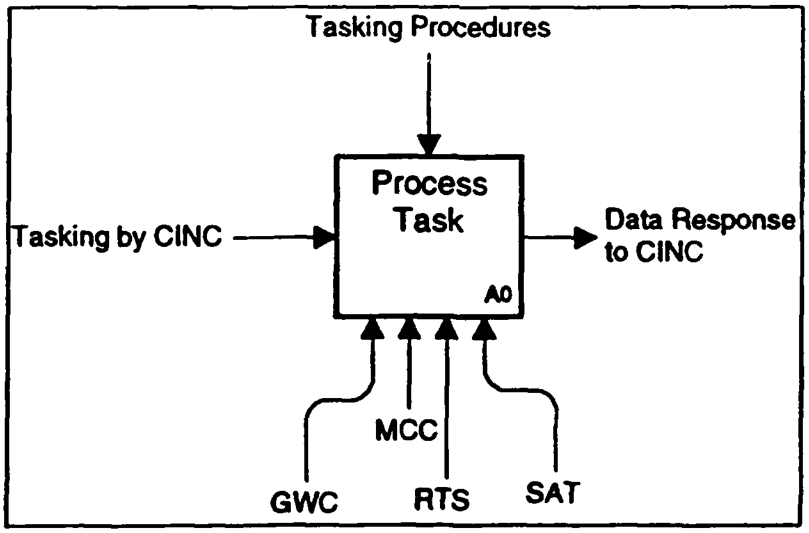
## 3.3 SIMULATION METHODS, ANALYSIS, AND RESULTS

### 3.3.1 Methods

The general approach was to simulate the flow of satellite support messages, responding to taskings from a warfighting CINC. Specifically, we were concerned with the time from when a field commander asked for new satellite special sensor information at a given location until he received that information. An assumption was made that the information required was not at the AFGWC.

The scope of the simulation was limited to 50 randomly generated requests for special weather data and focused on the top-level activities in processing the request from the time the request arrived at the AFGWC until the finished data were sent back to the JTF from the AFGWC.

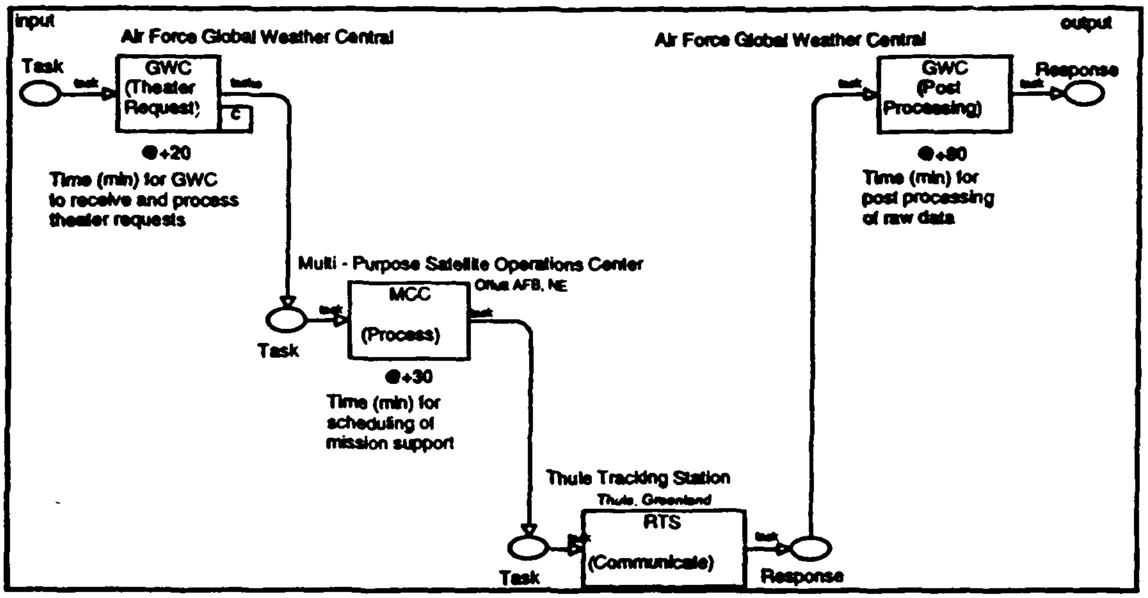
A top-level simulation diagram of the simulation is shown in Figure 3.2. The mechanisms shown are Global Weather Central (GWC), Mission Control Center (MCC), Remote Tracking Station (RTS), and the Satellite (SAT). In the simulation, these mechanisms are treated as subprocesses rather than resources.




**Top-Level Model**
**Defense Meteorological Satellite Program**
**Figure 3.2**

A decomposition of the activity *Process Task* in CPN language is shown in Figure 3.3. The time values shown are treated as constant and notional, based on data sources provided.

The focus of the simulation is on the Thule Tracking Station. This process was decomposed two levels further into subprocesses shown in Appendix B.




**Decomposition of Process Task**
**Figure 3.3**

The program structure and element data came from many sources, but primarily from the various element descriptions for the components of the system. Data concerning the amount of processing time for each function are *partially notional, and therefore resulting analysis may not be accurate.* One area for further development would be the incorporation of more accurate data regarding the processing time for each activity and subactivity to account for the transmission time between activities. Data concerning the tasking times and expected return times are not accurate but are sufficiently representative to illustrate the capabilities of the model. Also, station-keeping tasks on the satellites account for significant amounts of the communication time between the RTS and the satellites but were considered to be beyond the scope and interest of this simulation.

### 3.3.2 Analysis

As qualified above, *some data used in the analysis were notional.* Improved data could change the results; however, clearly the methods used are beneficial in analyzing the given problem. Though the results above are qualified, they indicate of the type of results that can be expected. The individual tasks were viewed in relation to the location of the satellites to determine hidden relationships between the length of time it took to process the message and the

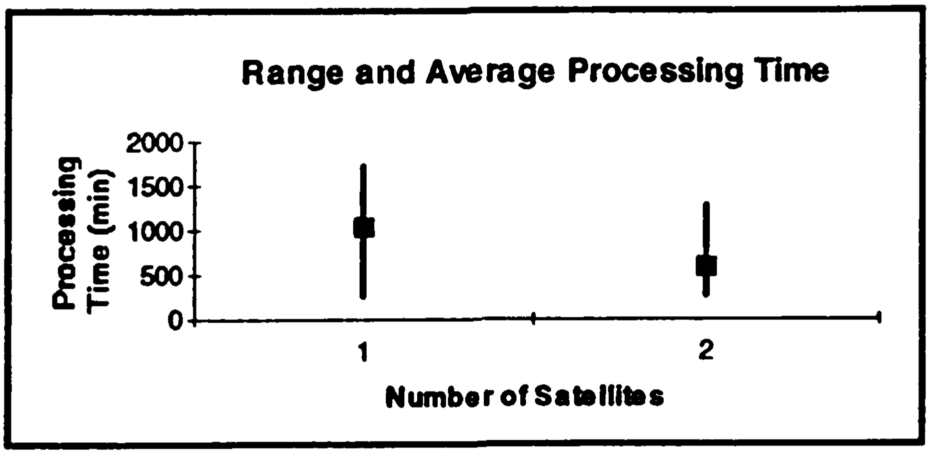
tasking. In this first approximation model, all orbits of the satellite come within range of the RTS; thus there was no discrimination for the collection and completion of the task based on the tasked location.

With two satellites in orbit, the first was 75° (not 180°) ahead of the second. As a result, the first satellite was tasked more frequently than the second. A constellation that had the satellites equidistant from each other would even out the tasking but would not address issues such as lighting. Currently both satellites track in the morning, one in the early morning and one in the mid- to late morning. As the orbits are sun-synchronous, they will continue to track over the same terrain at the same time each day with subtle shifts in the exact time due to jumps in the time zones.

Another aspect of the problem that would have had an impact on the results is the competion for scarce resources (e.g., RTS, MCCs, and communications links) with satellites that have other missions.

### 3.3.3 Results

The primary result of the simulation with the data used was that the addition of a satellite to a one-satellite constellation had a marginal impact on the processing time for satellite tasking by a Theater Commander. It would seem that by doubling the satellite capacity, response time could be reduced by 50 percent. However, this was not the case as illustrated by the actual results in Figure 3.4. The reason for the difference between expected and actual results was the slow processing time, attributed to operational processes on the surface of the earth involving satellite tasking and distribution of satellite products. Since satellites are expensive compared with ground support systems, funding priority should be given to ground support systems within ISCS.




**Range of Results for Differing Constellations**
**Figure 3.4**

## 3.4 OBSERVATIONS

### 3.4.1 General

As a validation effort for process simulation, Design/CPN proved to be a powerful tool in describing and analyzing command and control processes as they have not been analyzed in the past. It very effectively replicated the dynamics of large, detailed systems, showing bottlenecks in resource flow, idle activities, and providing end-to-end analysis.

### 3.4.2 Capacity Analysis

The Executive Summary of GAO report 92-3, "Military Space Operations: Satellite Control System Improved, But Serious Problems Remain, (U)," GAO/INTEL 92-3, 27 Dec 91, regarding the Air Force Satellite Control Network (AFSCN), states the following::

> **Capacity and Performance Management Program Inadequate**
>
> In order to measure computer use and performance and to help predict future needs, agencies should routinely collect and analyze detailed capacity and performance data. However, the Air Force is not doing this and, as a result, does not know how well [command and control segment] CCS is working and how much capacity is being used. Without this information, the Air Force *cannot* effectively determine whether and when changes are needed to meet mission requirements.
>
> Instead, the Air Force relies on three sources of information to manage CCS computer resources: (1) data on satellite contact success rates using CCS [current data system] and CDS, (2) computer operators' perceptions of CCS' limitations, and (3) infrequent ad hoc analyses of computer capacity and performance. While these provide some useful information, they do not give a complete picture of computer performance, mostly because they do not measure actual use or continuously assess performance. Furthermore, they do not offer the careful, comprehensive analysis needed to manage a system this large and complex.

The modeling effort described here could be extended to answer GAO's concerns. From its current state the model could be expanded in several directions. It could incorporate the entire DMSP network in a very detailed fashion to answer questions that are DMSP specific or questions concerning the entire satellite network. Typical questions that could be asked are: What happens to the response if a primary link is not available and alternate data routes are used? What will the loading on the alternate routes be if one or more of the satellites in orbit suddenly becomes inoperable? How long can the alternate MCC operate before becoming overwhelmed with tasking? At what point are the satellites overwhelmed? Beyond DSMP, the project could be expanded to answer capacity questions for all elements of ISCS.

### 3.4.3 Resource Allocation

The modeling effort could also look at the system from a different point of view. Instead of focusing on the turnaround time from a user, the maintenance cycles could be modeled to examine such elements as the conflict of resources between the various satellite programs, scheduled maintenance disruptions to the system, and intermediate stages in the system upgrade cycle. Alternatively, the system could be examined from a control standpoint, concerning transitioning partial control of certain subsystems from strategic to tactical control.

### 3.4.4 Interactive and Flexible Response to Simulation Customer

Each of the previously mentioned model expansion areas could provide valuable information. One of the great benefits to modeling with Design/CPN is the ability of the analyst to work with the system experts and to apply that synergism directly to the model. Clients get exactly the types of information in which they are interested. Also, they can have a model developed very quickly with the interactive capability provided by Design/CPN. For any effort or direction of expansion, the largest portion of time and the greatest amount of effort would be devoted to the collection and correlation of relevant and specific information.

# 4.0 THEATER MISSILE DEFENSE

## Counterforce Options

## 4.1 BACKGROUND

In this effort, ANSER was tasked to demonstrate the capability of modeling and analyzing differing rules of engagement (ROEs) or decision-making strategies. The Strategic Defense Initiative Organization (SDIO) selected the Counterforce Options problem in Theater Missile Defense (TMD). Counterforce is reacting to the enemy's missile launch capability. Procedurally it is broken down into detection, decision, action and evaluation activities. Appendix C contains diagrams of Theater Missile Defense.

ANSER's research surfaced the Phase One Engineering Team's (POET's) study of Theater Missile Defense. This study included an IDEF0 model of the Counterforce Options problem. In order to constrain the level of effort to the funding level provided, ANSER, in agreement with the government, limited analysis and modeling to the portions of the existing study relating to pre-launch detection of an enemy missile launcher, i.e., cases where a launcher was detected before launching a missile. Also, the scope of the project limited consideration of enemy air defenses and postexecution weapon failure. Follow-on work could include the addition of these aspects, which would be a relatively simple application from the developed simulation.

## 4.2 OBJECTIVE

The objective of the simulation was to develop a tool for evaluating alternative rules of engagement. Of particular concern was comparing rules to assess the best use of scarce and valuable competing resources for attacking a theater missile launcher.

## 4.3 SIMULATION METHODS, ANALYSIS, AND RESULTS

### 4.3.1 Methods

The general approach was to use IDEF0 in combination with colored petri nets (CPNs) to facilitate development of a quick-response "what if" model. In order to evaluate timing issues or to analyze the effects of changes (particularly nonlinearities) in the system, a CPN simulation can be created. If the IDEF0 model is validated, and the activation rules are accurately represented, then the output from the simulation can be used to evaluate different ROEs. Some scenarios can be tested by simply changing input files to the model: different weapon mixes, sensor placements, or weapon attributes. Other "what if" scenarios require more significant changes in the code of the simulation, but can provide good evaluation on the effects of changing rules or methodologies. Due to time constraints, ANSER looked more closely at the second of these methods, as more significant differences could be analyzed.

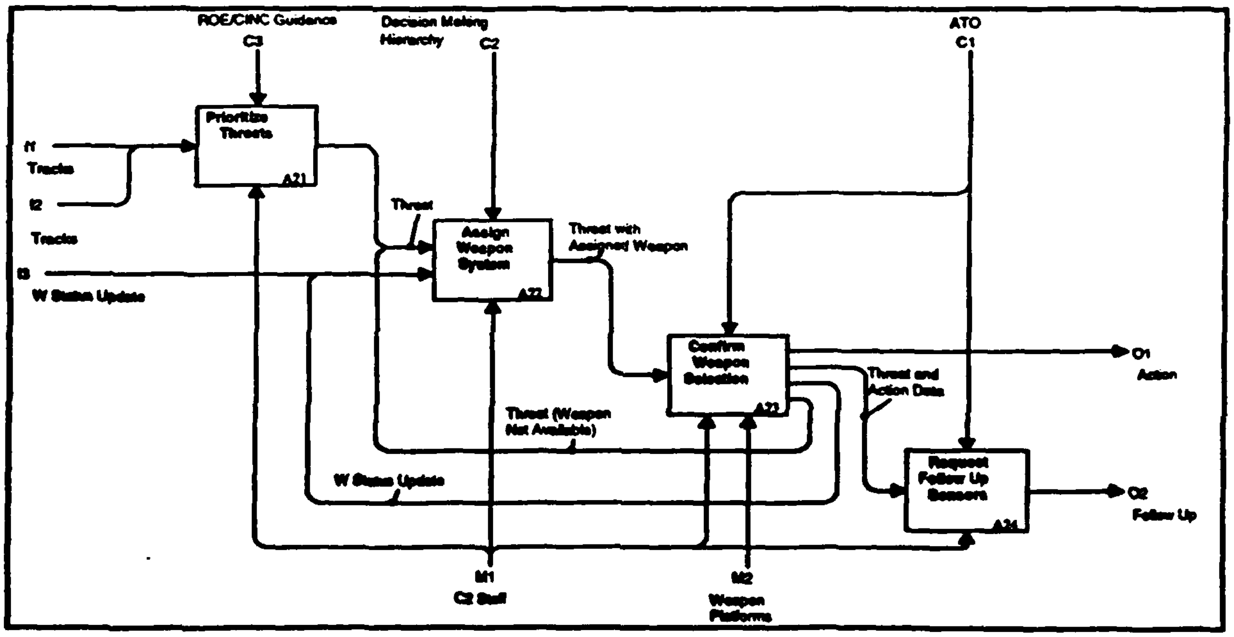
IDEF0 is a standard language for describing systems. We were able to take an existing IDEF0 description of the overall problem of Counterforce Options and tailor it to meet our proof-of-concept needs. This eliminated the need for duplication of research and modeling effort. We were able to input the earlier work into Design/IDEF, an automated tool supporting the methodology. IDEF0 describes systems in terms of activities and the connections between them. The information and resource need for each activity is also described.

In our effort to reduce the scope of the model and keep it unclassified, we limited ourselves in several areas. As mentioned above, the model only simulates the premissile launch capabilities. Other limitations include a restriction to generic weapons, targets, and sensors. The four types of generic weapons are

- *Army Missiles*: All Army surface-to-surface missile systems
- *Navy Missiles*: All surface-to-surface missiles fired off Navy ships
- *Navy Aircraft*: All planes assigned to TMD from Navy carriers
- *Air Force Aircraft*: All Air Force planes assigned to TMD.

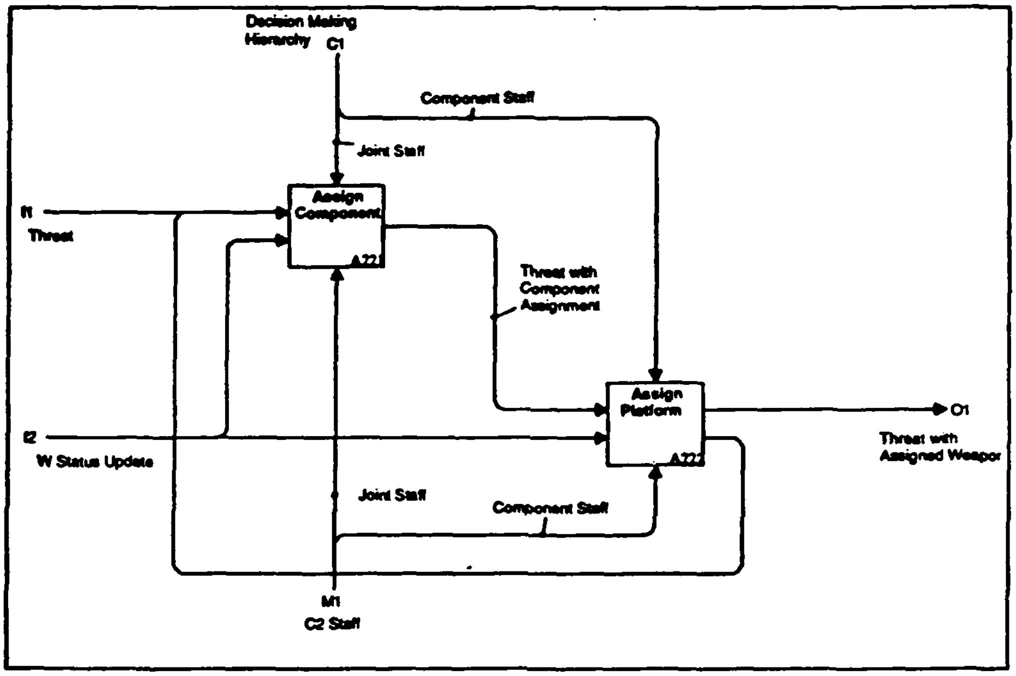
All threats were categorized as generic enemy missile launchers. Sensors were assumed stationary, and each covered a fixed portion of the area of operations. The simulation was designed to accept actual parameters and other details of the systems used for these operations if the government required it.

In order to meet the objectives, ANSER modeled two *scenarios*. The *difference* between the two is in how weapons are controlled.




**A2 Node for Both Scenarios**
**Figure 4.1**

Scenario "A" was defined by having TMD counterforce weapons assigned to subordinate commands. In the activity model, Scenario A is distinguished by the decomposition of the A2 node (Figure 4.1). In Scenario A, threat notification is given to the Joint Headquarters. The threat is then assigned to a specific component based on geographical location of the threat and areas of operation (decomposition of A22, Figure 4.2). The component commanders then assign specific weapon platforms to neutralize the target (A223).




**A22 Node for Scenario "A"**
**Figure 4.2**

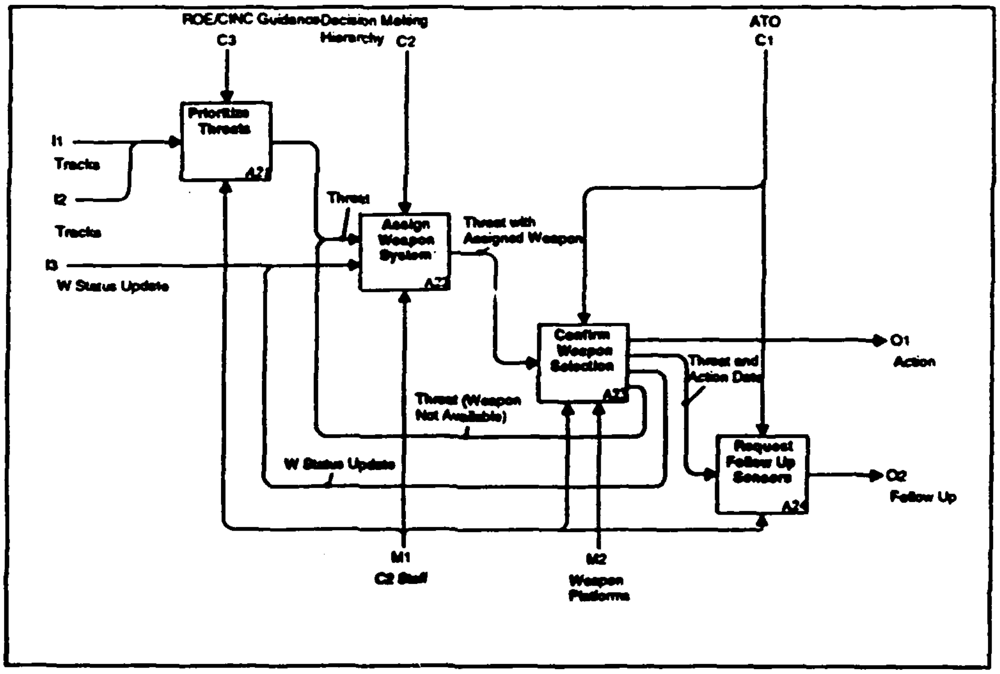
Scenario "B" was defined by having all weapons assigned to Counterforce directly under Joint Command. Therefore, specific weapons are assigned in box A22 without need for decomposition.

Some analysis can be done with the activity model (IDEF0). It is obvious that the communications delay will be less in Scenario B, as one participant is effectively removed from the loop. Communications requirements to the Joint Commander are increased, however, as he must be aware of the exact status of each weapon platform. The driving factor in choosing between these alternatives will mostly likely be how the rules affect success ratios rather than information requirements. In order to provide this type of analysis, it is necessary to move into dynamic modeling or simulation.

Simulation tools are available for further analysis. One simulation approach is to use CPN that support the IDEF0 structure. The mathematics of CPN is powerful in its capability to analyze nonlinear phenomena. The two building blocks in IDEF0, *arrows* and *activity boxes*, map directly to *arcs* and *transitions* in CPN. A third CPN construct is a *place* that corresponds to queues between activities. Figure 4.3 shows an IDEF0 page and Figure 4.4, its corresponding CPN page.

ANSER used Meta Software's Design/CPN to automate the simulation processes. The conversion between IDEF0 and CPN is computer assisted, as these tools are built by the same developer. Terms defined textually in IDEF must be defined in code for CPN. Activation rules, which in IDEF0 define how activities transform their inputs into output, must also be encoded. The programming language used is Meta Language (ML), a LISP-like language.

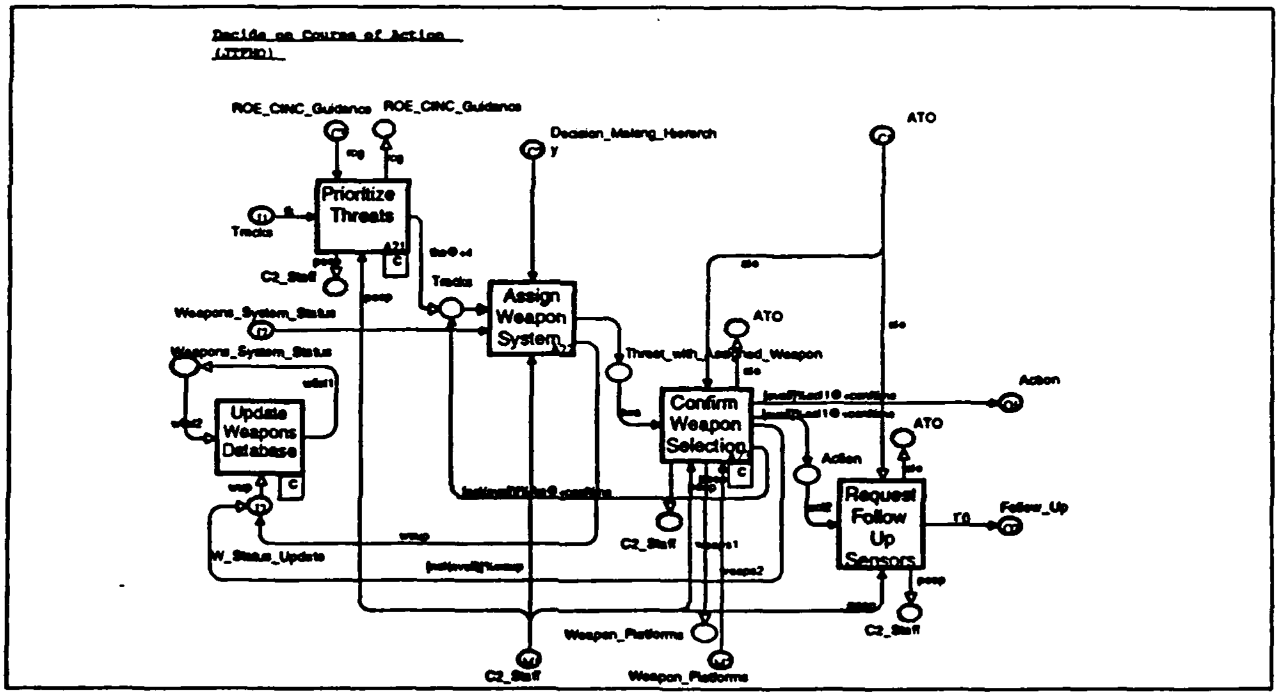
Along with code for IDEF's text, there are two other types of functions that must be added as activities into the model. These are the development and maintenance of the data, normally outside the scope of the IDEF0 model.




**IDEF0 Page**
**Figure 4.3**

While an activity model may assume updates occur in a timely and accurate manner, in order to simulate, it is necessary to maintain a database so that each decision is based on current data. An activity, *Update Weapons Database*, is added to perform this maintenance (see Figure 4.4). We call these "bookkeeping" activities and add them where necessary. These also

include initialization activities, which give variables their initial values. The other new activity type could be called *Analysis*. These activities collect or display statistical information on the simulation run.




**CPN Page**
**Figure 4.4**

Once the additional activities have been incorporated and all necessary code has been written, the simulation can then be run with different input values to analyze the effectiveness of each scenario under differing conditions. Design/CPN allows for repeatability of results by requiring the user to set the "seed" for the random number generator. If the seed is the same and the initial conditions are the same, then the results will be identical. This makes testing of different scenarios on the same input statistically significant, as it allows the scenarios being tested to face the same exact situation. In order to obtain statistically significant results over a sample of random initial conditions, it is necessary to run both scenarios on the same input with a set of random seeds.

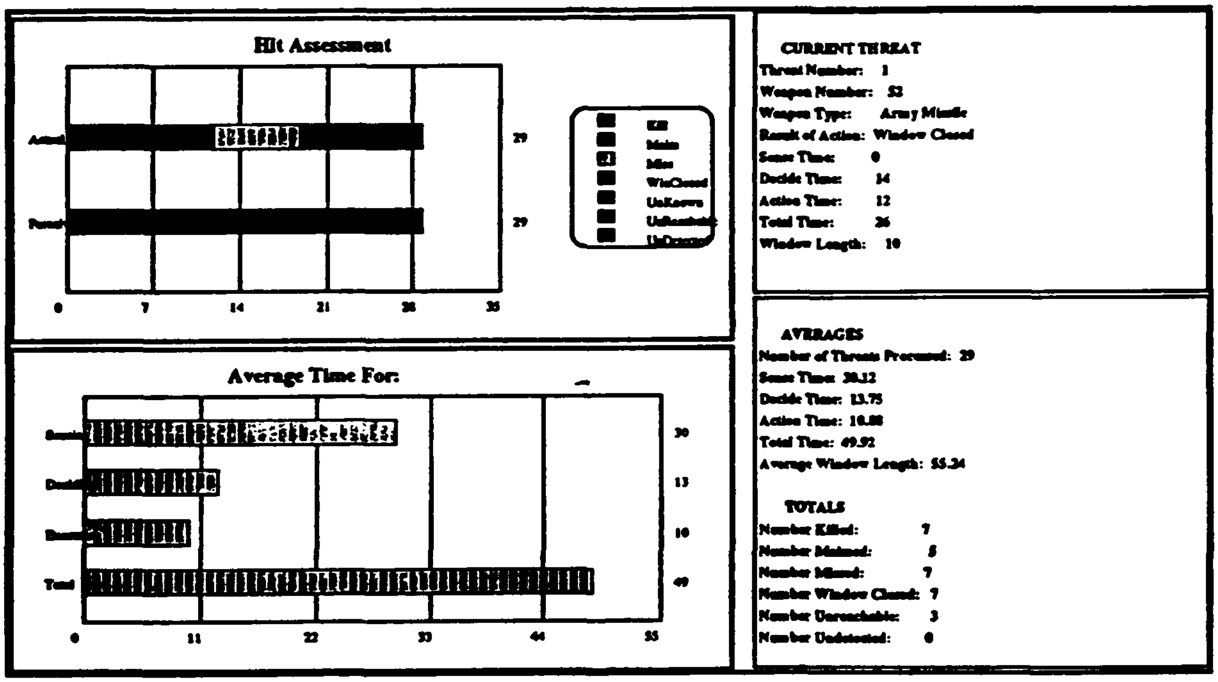
ANSER was required to use *notional data* in performing analysis on a proof-of-concept model. Results obtained are "real" in that they represent actual output of simulation runs, but are "notional" in that the simulation relied heavily on notional input.

In order to evaluate the relative effectiveness of the two scenarios, we monitored the usage of the different weapon types and the success ratio. We defined the outcome of a strike to be *Kill, Maim, Miss,* and *Window Closed.* Because targets did appear out of sensor range as well as out of range of available weapons, we also included *Undetected* and *Unreachable.* The

success ratio was the sum of all hits (*Kills* + *Maims*) over the total number of strikes. A *Window Closed* result occurred when the strike did not reach the target before a predetermined amount of time had elapsed. *Window Closed* represents a missile that has already been launched or a situation where the launcher has been moved, i.e., the target no longer exists at the location specified.

### 4.3.2 Analysis

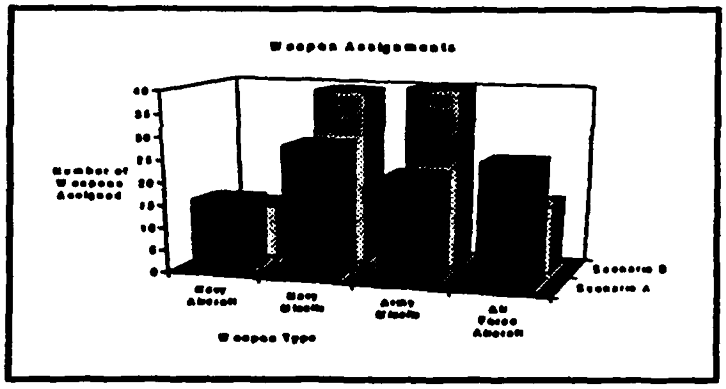
Figure 4.5 shows the output of one CPN simulation run. These four charts are dynamically updated while the simulation is running. The chart on the top left shows the actual and perceived results of a strike. Since a missile, unlike a pilot, cannot *perceive* its effect, the *Unknown* category was added.




**CPN Statistical Output**
**Figure 4.5**

The outcome of a strike is determined by combining the accuracy and lethality of the weapon with the surety of the target information. The chart in the lower left keeps cumulative averages for the time spent in each of the three stages: Sense, Decide, and Execute, as well as a total average time. The text chart on the top right has the statistics on the current target, which change during simulation. It is possible to note multiple strikes on the same target, as well as any abnormalities in time values. The lower right is a textual representation of the numbers graphed in the other charts. While these charts are not particularly helpful for statistical analysis, they can be used to detect run-time problems while the model is being developed or tested.

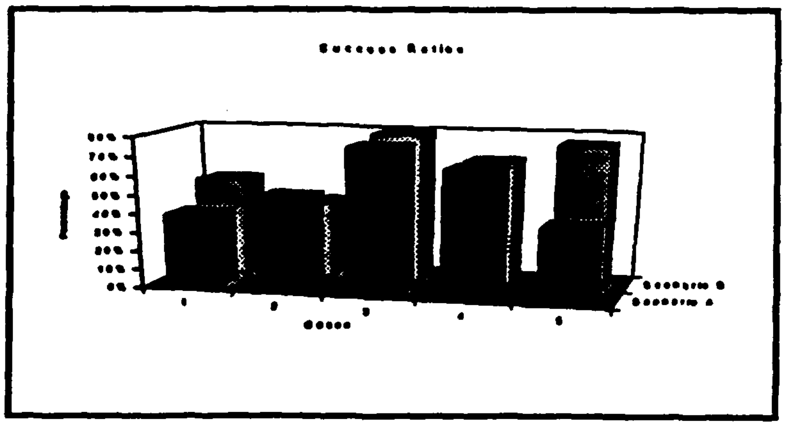
Information can be manipulated on the spreadsheet Excel to produce more useful charts for analysis. Outputs from the simulator can be exported as ASCII files after each run to any one of a number of analysis software packages. The analysis can be done in Excel, comparing output from different scenarios and different random seeds. Figure 4.6 shows the numbers of each type of weapon assigned in each scenario over five sets of runs.



**Weapon Assignments**
**Figure 4.6**

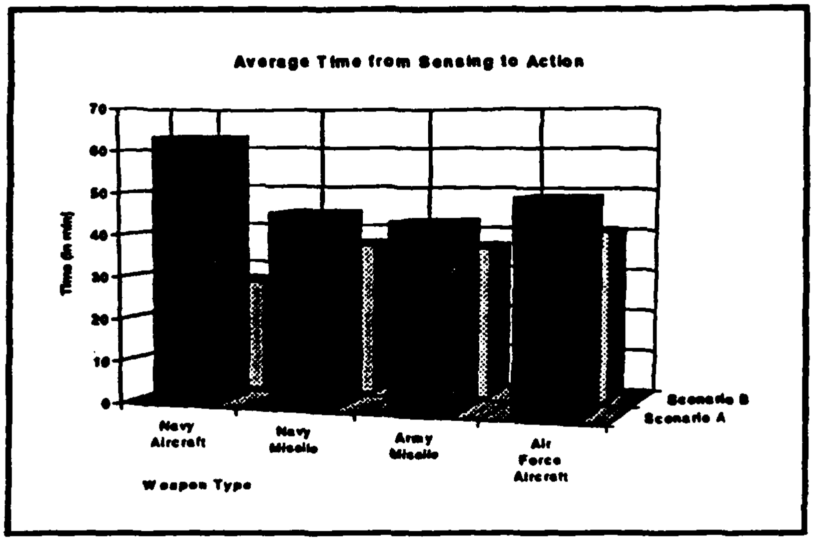
Scenario B uses fewer planes because the algorithm for selection is "fastest steel on target" without regard to weapon type. Since missiles are twice as fast as planes, a missile will be selected unless the target is next to an airfield or a coast by a carrier. We did not incorporate Combat Air Patrol (CAP), so this result is skewed, reflecting the limits to the analysis imposed by notional data.

Figure 4.7 shows the success rates (defined by the ratio of *Kills* plus *Maims* to total strikes) of each of the five runs of each scenario.

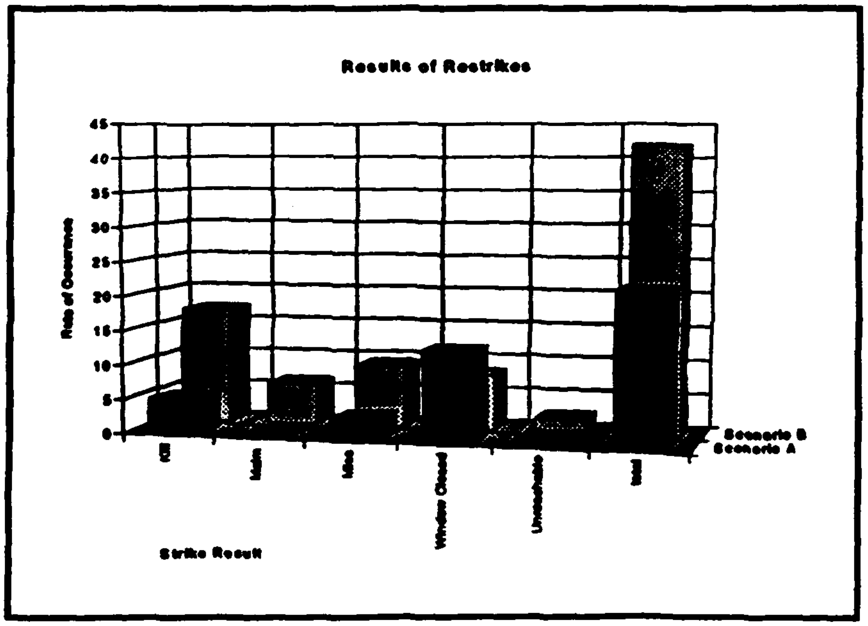


**Success Ratios**
**Figure 4.7**

The success ratios differ in part because of weapon choices. Scenario B had fewer *Window Closed* outcomes because more missiles were chosen, which are twice as fast as pilot-drive aircraft. Other factors include the average time (Figure 4.8) and the number of restrikes possible (Figure 4.9).




**Average Time**
**Figure 4.8**




**Restrike Results**
**Figure 4.9**

More restrikes are possible in Scenario B because of the higher rate of missile selection and the decrease in time used for a first strike. More kills are attained because the "lethality" of a missile was entered as greater than that of planes. What was not incorporated, however, was the fact that if surety is decreased, a plane will be more effective than a missile due to the pilot's ability to modify target coordinates based on visual observation.

The combination of IDEF0 and CPN is especially powerful because of the many aspects of nonlinear systems that can be accurately modeled. As a standard language for describing systems, IDEF0 promotes the reusability of models. However, an IDEF0 model cannot be simulated immediately if the appropriate data have not been collected. When building an IDEF0 model that might be converted to simulation, it is vital to collect very specific activation rules and durations for the activities. It is also necessary to document thoroughly the mechanisms of processes. Each mechanism should have efficiency, accuracy, and failure-rate information at a minimum, along with any other information necessary to model its effect on the activity.

The above would be necessary for conversion to any simulation tool. The transition from Design/IDEF to Design/CPN is partially automated. The software checks for any obvious ambiguities, e.g., arrows, branches, or joins, and prompts for the resolution. It also creates the beginnings of the code necessary for simulation by listing, with appropriate syntax, the data types used in the IDEF0 model. In short, it creates the prototype for a Global Declaration Node in CPN. As activation rules in IDEF0 are strictly textual, there are no means of automatic conversion to CPN, but the information must be there for the modeler to be able to code system behavior.

A purpose of modeling systems is to document existing practices, test proposed solutions to existing problems, and model future systems, identifying strengths and weakness prior to implementation. In business applications, a dominant measure in comparisons is cost. In modeling C2 systems, cost may also be important, but overriding measures are effectiveness, expediency, and accuracy. Therefore, timeliness of information flow and effectiveness of different systems are frequently the focus of models of C2 systems.

An activity model (IDEF0) can show *disconnects* in systems where information is not arriving at the required destination. It cannot show *when* the information arrives if a connection exists. A simulation can resolve an activity model to its lowest level and in bits per second provide the data needed for analysis of timeliness. A complete process simulation can show where information is being delayed, when it is being received, and the effects of delay on mission success. It can also show what processes or equipment are not being used efficiently by a system (i.e., frequently idle). A simulation can relate delays and idle processes to the impact of specific information. For example, while it may not be acceptable for a commander of a flight to have sensor threat information that is an hour old, it may be acceptable for planners of the next day's battle.

Simulation is also essential to discussing the impact of differing ROEs, as in this model. The differences in the IDEF0 model did not show the effect of varying methods on the mission.

Running a simulated version of the model allowed comparisons of the methods under different circumstances. This is an area where CPN is especially strong. The flexibility of CPN allows nonlinearities to be accurately portrayed so that counterintuitive results can be documented and analyzed.

One means of analysis is the use of the real-time charting facilities provided within Design/CPN. As shown in Figure 4.5, the capability has limitations, but it can be useful in dynamic analysis, providing the capability of observing how backups build and are dissipated, or in this model, how often a particular threat was targeted and the results of the restrikes. This makes it possible to vary input parameters to see which processes are directly affected and will fine tune the system.

### 4.3.3 Results

The result of comparing Scenario A (TMD weapons assigned to subordinate command) and Scenario B (TMD weapons assigned directly to Joint Task Force) was that Scenario B was more effective in terms of target kills and resource management.

The project demonstrated a simulation capability for assessing doctrinal issues associated with weapon system development or use. Prior to this demonstration, the SDIO acknowledged that no other tools have been used that were effective in this type of analysis.

## 4.4 OBSERVATIONS

### 4.4.1 Doctrine Analysis

The project showed the capability to analyze how changes in ROE or decision-making hierarchies affect the success of a combat operation. We can compare hypothesized methods or systems prior to implementation or acquisition in order to analyze effectiveness and efficiency.

### 4.4.2 Action Modeling and Simulation

The ability to take existing models in a standard language (IDEF0) and port them with relative ease into a simulation environment eliminates the need for duplication of research before using simulation techniques for analysis.

# 5.0 DISCUSSION

## 5.1 GENERAL

This section contains some general observations that can be drawn from all of the command and control projects. It was ANSER's assessment that objectives set for each project were met and that interesting conclusions could be drawn on the benefits of simulation to the subjects of these projects. At a different level of discussion, it seems important to question how these projects impacted the corporate information management (CIM) initiative in general.

## 5.2 POLICY ISSUES

While some information is available within the DOD literature (directives, guidebooks, etc.) on process simulation, it is generally incomplete as to how process simulation fits in with other CIM process simulation methodologies.

## 5.3 FUNCTIONAL PROCESS IMPROVEMENT

Relationships between process simulation and other functional process improvement methodologies were determined.

- Simulation provides a temporal perspective to process modeling
- Simulation can quantify costs and benefits of processes. This capability can support functional economic analysis
- Simulation provides a means of measuring the value of improvements expected by "to-be" model candidates
- Process simulation validates activity and data modeling. Data collection required for simulation can most efficiently be done during activity modeling. Through collection of this data, activity modelers are disciplined into defining processes more rigorously Similarly, data structures required for simulation are precisely the data structures required in data modeling. Process simulation is where the essential elements of activity and data modeling are replicated. In addition, simulation addresses dynamics.

## 5.4 ANALYSIS ISSUES

Process simulation provides a means of analyzing and assessing temporal issues in processes. Simulation can reveal

- Utilization rates of activities (bottlenecks and idle activities)
- Global value of local changes within organizations.

Data for process simulations should be collected during activity modeling for two reasons:

- Data collection for simulation forces activity modelers to understand processes better
- Efficiency. The activity modeling includes collection of data about processes. Why not include temporal data about processes and save a follow-on collection effort?

## 5.5 DATA COLLECTION

Generic data required for simulations include

- Procedures for defining a process cycle, including the rules for initiation, control, and termination of a cycle
- Time required for each cycle
- Number of cycles required of a child process for each cycle of its parent process
- Quantity and value (cost and utility) of resources consumed by a process.

## 5.6 SIMULATION TOOLS

The projects studied in this task were all produced using Design/CPN, a commercial off-the-shelf (COTS) package. Design/CPN features the following properties.

- Colored petri nets (CPNs) are analytically sound in dealing with difficult problems of command and control such as parallelism, synchronicity, and conflict for resources. As in activity modeling using IDEF, CPNs are hierarchical. This is an important property in that few simulation tools have comparable capabilities in addressing complexity typical of information systems.
- Design/CPN is object oriented and consequently can be applied with minimal programming using an excellent user interface.
- An automatic program feature of Design/IDEF converts IDEF0 diagrams to CPN diagrams for Design/CPN.

Other tools considered for comparison purposes were MODELER, an Air Force-developed petri net simulation software similar to Design/CPN; ITHINK, a COTS simulation software using a simulation approach known as systems dynamics, and SIMPROCESS, a COTS simulation software using a SIMSCRIPT programming approach. Some observations about these tools are noted below.

- MODELER is a powerful tool similar to Design/CPN developed by the Air Force. It has a more user-friendly interface, is limited to the SUN workstation, but is not integrated with IDEF

- ITHINK is a powerful tool for small systems lacking hierarchy. It is an inexpensive COTS tool

- SIMPROCESS is an interesting tool with a graphical user interface, using familiar icons to represent system components. SIMPROCESS appears not to have the analytical power of the other tools we considered, especially in addressing nonlinearities. It also lacks hierarchy.

## 6.0 FINDINGS

**6.1** All of the projects were successful in meeting the objectives established for them. Collectively they produced the following findings:

There is a need to develop DOD policy and procedures on the application of process simulation.

Relationships between process simulation and other functional process improvement methodologies were determined.

- Simulation provides a temporal perspective to process modeling
- Simulation can quantify costs and benefits better than other means for some processes, such as those involving resource flow and those that are in transition. This capability affects significantly functional economic analysis.

**6.2** Process simulation provides a means of analyzing and assessing temporal issues in processes. Simulation can reveal

- Where bottlenecks occur in resource flow through a process
- Idle processes
- Global value of local changes.

**6.3** The simulation projects accredited Design/CPN as an effective tool in analyzing processes. Some interesting features of this tool included

- Object-oriented design features that minimize programming and facilitate the construction of a simulation in a logical way
- The graphical interface, facilitating setup and analysis
- Powerful analytical capability. Petri nets were designed to address complexity inherent in information management. For instance, they deal with sychronicity, parallelism, and conflict for resources,, nonlinear concepts that are difficult for other simulation techniques
- The automatic program features of Design/IDEF that convert IDEF0 diagrams to CPN diagrams for Design/CPN
- Reusability of simulations. Simulations can be easily perturbed for sensitivity analysis or modified for excursion.

**6.4** Data for process simulations should be collected during activity modeling for two reasonS:

- Collection of data for simulation forces an activity modeler to better understand the processes in the activity model
- Collection of simulation data during activity modeling assures continuity of analysis between activity models and associated simulations.

**6.5** Generic data required for simulations include

- Procedures for defining a process cycle, particularly initiation, intention, and termination control of a cycle
- Time required for each cycle
- Number of cycles required of a child process for each cycle of its parent process
- Quantity and value (cost and utility) of resources consumed by a process.

## 7.0 RECOMMENDATIONS

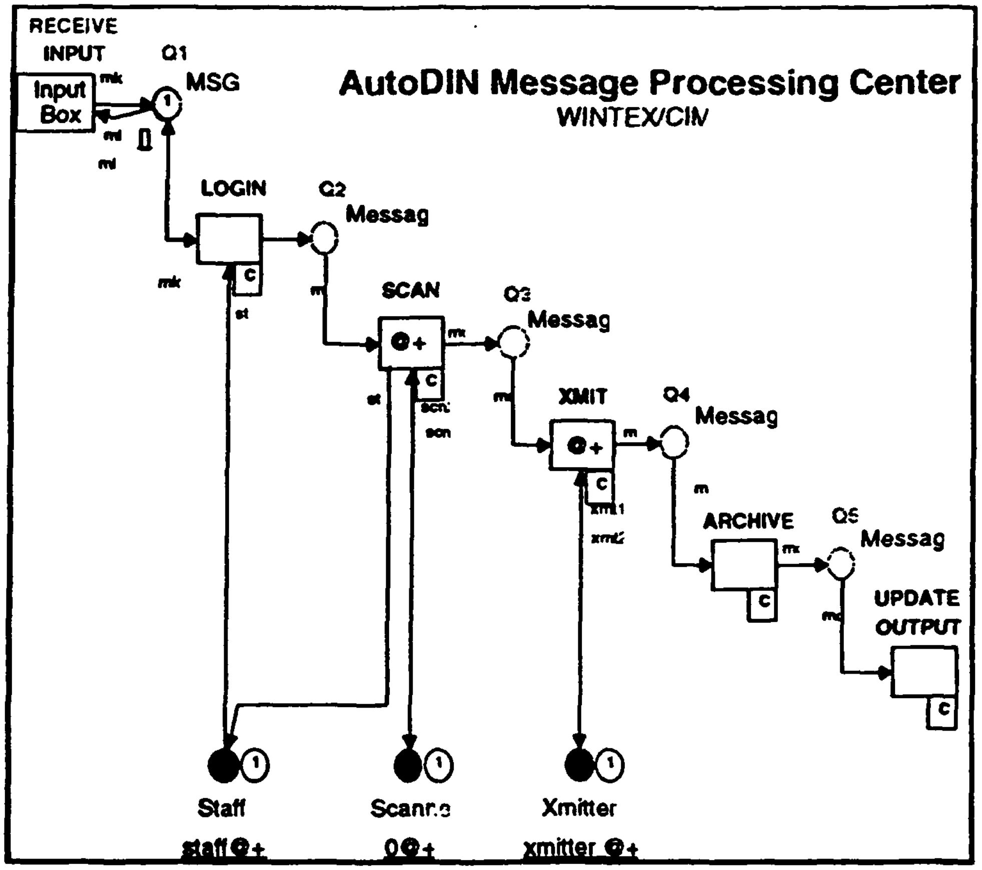
Two recommendations result from this effort:

**7.1** The Director of Defense Information should develop and publish procedures for applying process simulation as a CIM process improvement methodology. Procedures should focus on temporal issues and should include itemizing data collection during activity or data modeling that will facilitate process simulation.

**7.2** The Office of the Secretary of Defense should conduct independent verification, validation, and accreditation to the extent feasible of process simulation tools (e.g., Design/CPN, MODELER, ITHINK, SIMPROCESS, PACE, and others as appropriate).

# APPENDIX A

# DIAGRAM OF MESSAGE PROCESSING CENTER




**Receive Input:** This node is where messages generated by the C2 staff are placed for processing by the MPC staff. The dual arrows to the Q1 and the dual arrows from Q1 to **Login** enable the messages at Q1 to be sorted by message priority. Priority one messages are generated every 16 minutes, while priority two and three messages are each generated every 14 minutes each.

**Login:** This node is where the MPC staff starts processing the message. The oldest message of the highest priority is chosen from Q1. Information regarding the message is then entered into a log book kept by the MPC staff. At this node, the MPC staff member also proofreads the message to ensure it follows conventions and that all of the required entries are made regarding message priority and handling instructions. This process takes an average of 2 minutes for priority two and three messages and an average of 4 minutes for priority one messages. Note that the MPC staff member is not released at the end of the process (arrows are not dual-headed) but is "carried" with the message to the next process.

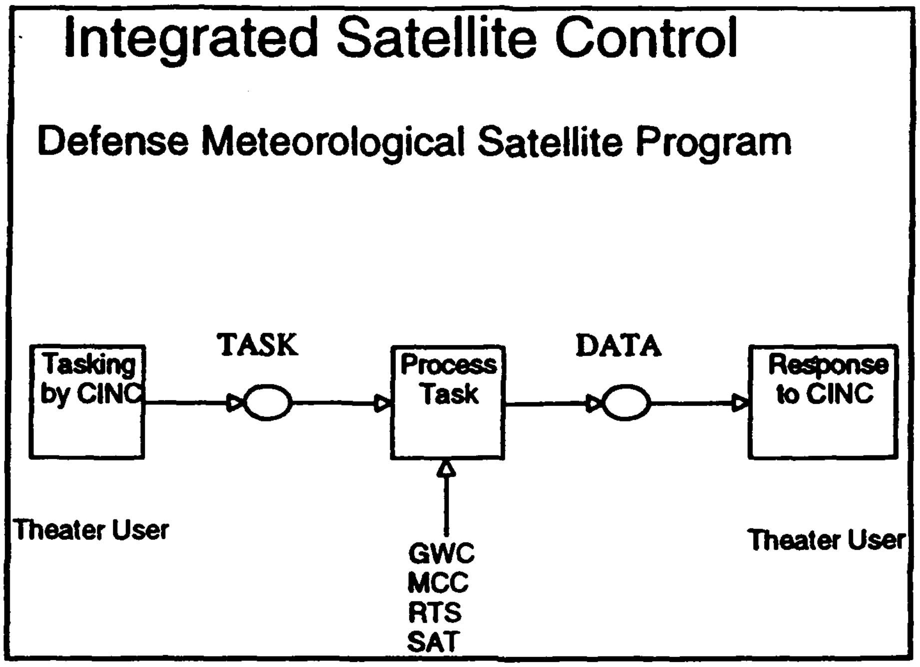
**Scan:** The message and staff member enter this node only when the scanner is not busy processing another message. The scanning process takes an average of 5 minutes regardless of message priority. When the process is finished, the MPC staff member is released to go back and process another message at **Login**, and the scanner is returned to the idle state. The message is sent to Q3, where it waits to be transmitted.

**Xmit:** This is where the now electronic message is transmitted over the AutoDIN network. The process takes an average of 1 minute regardless of message priority and requires the xmitter (transmitter) be idle to initiate. No MPC staff involvement is required for this process.

**Archive and Update Output:** Once the message has been transmitted, it has passed out of the simulation. These final two nodes are used to output information regarding the message throughput time to an ASCII file and to update the simulation graphs showing backlog and information about the idle time of the scanner.

# APPENDIX B

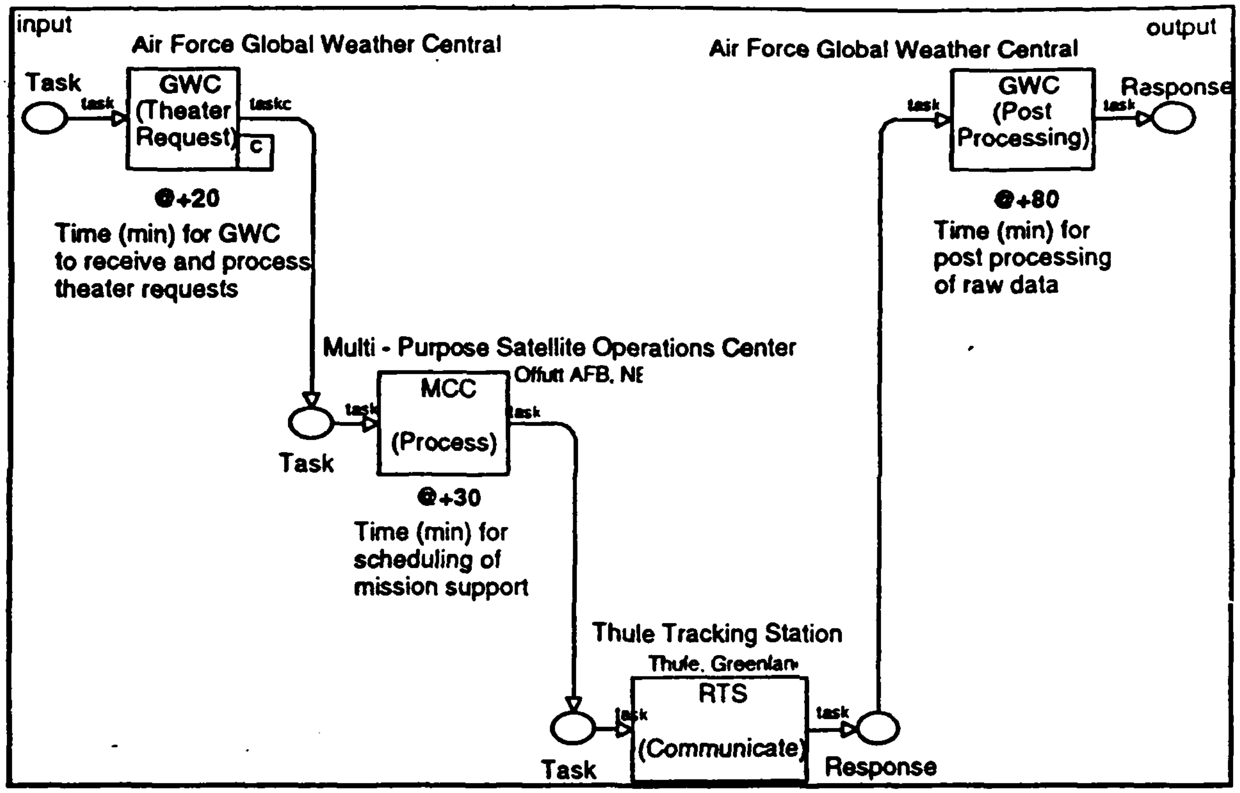
# DIAGRAMS OF INTEGRATED SATELLITE CONTROL SYSTEM




**Tasking by CINC:** At this node, the requests for special sensor data are randomly generated, and the initial framework for the simulation graphs are created. Fifty requests are generated, which can range from 0° to 359° latitude and 0° to 359° longitude with a requested response time ranging from 3 to 6 hours. Each message has a generated start time ranging from 0 to 120 minutes after the previous message.

**Process Task:** This node represents all of the processes needed to produce the data required by the CINC. This box is decomposed on the next graph and will be discussed in further detail there.

**Response to CINC:** This node represents the data gathered in the previous node being returned to the CINC. Also in this node are regions to update the simulation graphs, providing information about response time for each task. Finally, the information regarding the task and response is recorded in an external ASCII file for outside analysis.

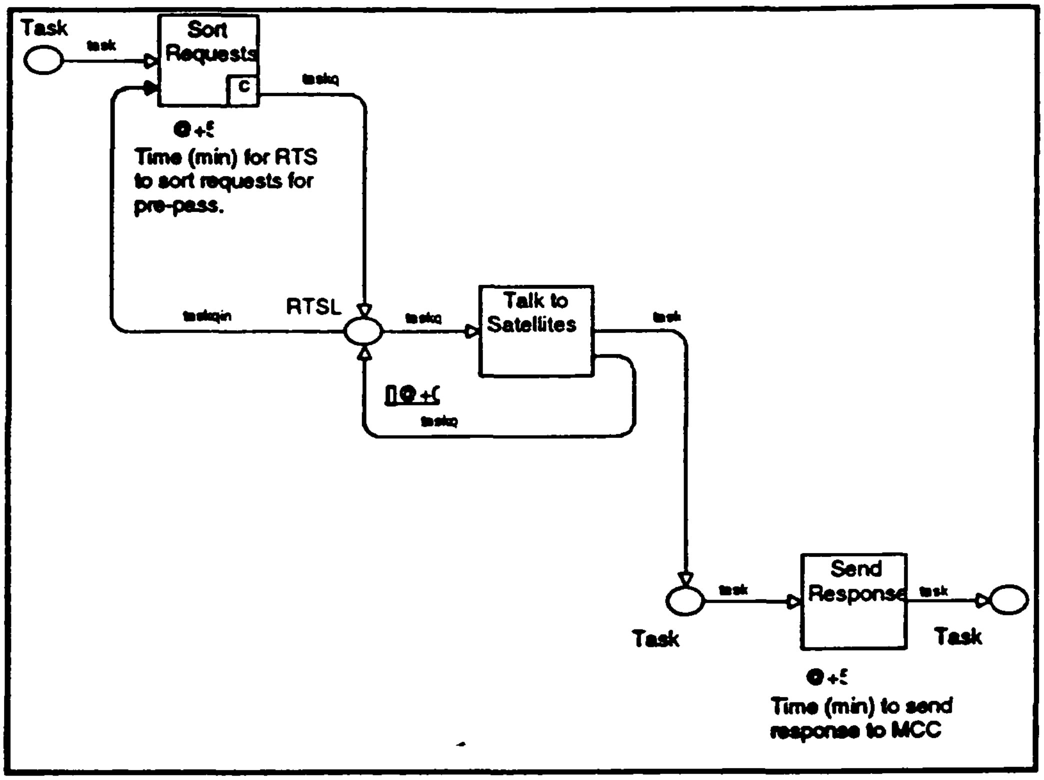



**Air Force Global Weather Central (left):** This node represents the activities that take place at the AFGWC when a task is received from the CINC. All communication with the CINC and representatives takes place through the AFGWC. This node has not been decomposed because of the limit to the scope of the simulation and because detailed information regarding the operation of the AFGWC was not obtainable within the limited timeframe of the simulation effort. A notional time of 20 minutes was assigned to the AFGWC for initial processing of the data tasking.

**Multipurpose Satellite Operations Center (MPSOC):** This node represents the activities that take place at the MPSOC. The task is forwarded to the MPSOC from the AFGWC. The primary activity here is the scheduling of the task response on the satellite and the communication with the Remote Tracking Station (RTS), which will actually communicate with the satellite. Other activities are the analysis of telemetry and tracking data to determine the health and status of each satellite and the initial testing of new satellites in orbit. This node has not been decomposed, as it was deemed to be beyond the scope of the current proof-of-capability demonstration. A notional time of 30 minutes was assigned to the MPSOC to represent the time spent scheduling the communication between the RTS and the satellite.

**Thule Tracking Station:** In this simulation, a single RTS was used, Thule. This node is decomposed to show in greater detail the activities that take place in the communication between the RTS and satellite. When the data are returned from the RTS, they are given to the AFGWC for final processing and forwarding to the CINC.

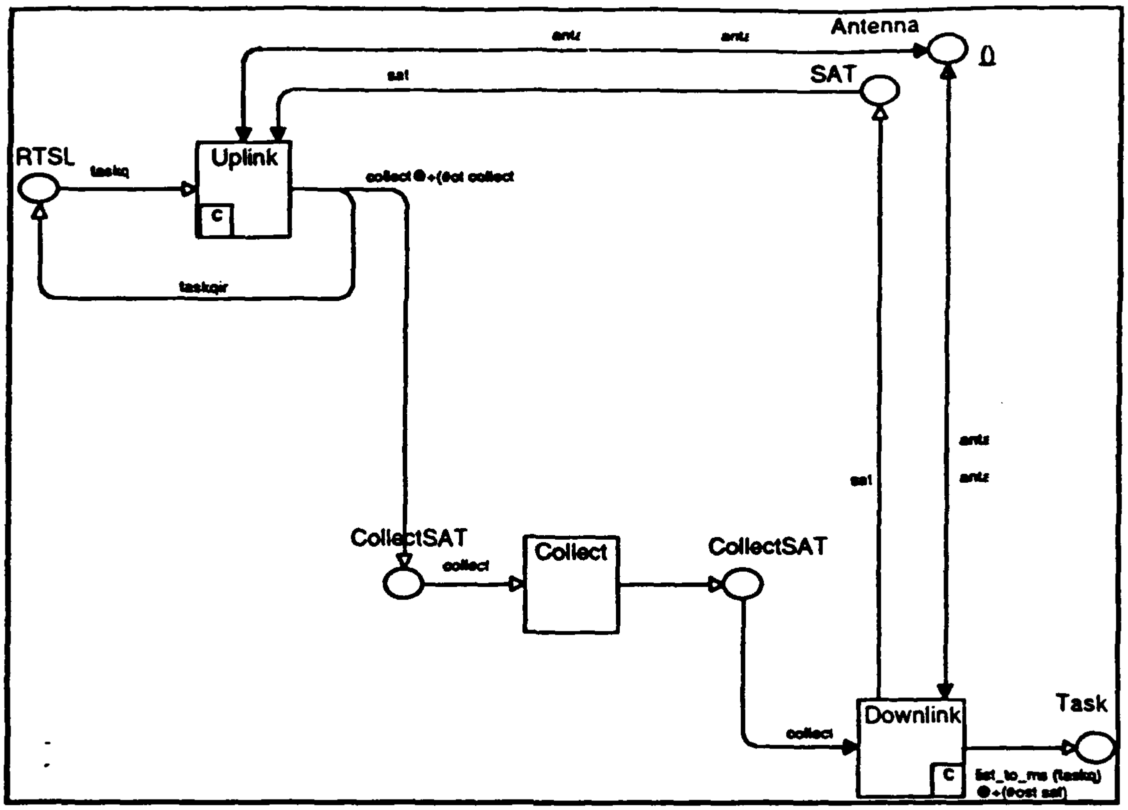
Air Force Global Weather Central (right): When the RTS returns the data, the AFGWC archives the data, processes the raw data into detailed information and intelligence, and passes the final product on to the requesting CINC.




**Sort Requests:** Once the task is received by the RTS, it is sorted and held waiting until just before the satellite comes into view again. At that time, the station is prepared for communications with the satellite (called prepass preparations). During the pass, the RTS has approximately 15 minutes to communicate with the satellite. A notional time of 5 minutes is assigned to the RTS for prepass preparation.

**Talk to Satellites:** This node has been decomposed to show the details of the communication with the satellite during the pass period.

**Send Response:** Once the data are received from the satellite, they are bundled and transmitted back to AFGWC. A notional time of 5 minutes was assigned to the RTS for this postpass processing.




**Uplink:** This node represents the RTS uploading new commands to gather data to the satellite. To perform this process, the satellite has to be visible to the tracking station antenna, and the antenna initially has to be idle. A time limit of 15 minutes based on the satellite orbit was imposed to complete both the uplink and downlink processes.

**Collect:** During the rest of the satellite's orbit when it is not visible to the RTS, it is collecting data according to the uplinked command, and storing them for downlinking on the next pass over the RTS. The satellite orbit of 101 minutes dictates the time allocated for this activity.

**Downlink:** On a pass over the RTS after the data have been collected, they are passed down to the RTS where they can be forwarded to AFGWC for postprocessing and finally shipped out to the CINC. A time of 15 minutes based on the satellite's orbit was imposed to complete both the uplink and downlink processes.

# APPENDIX C

# DIAGRAMS OF THEATER MISSILE DEFENSE

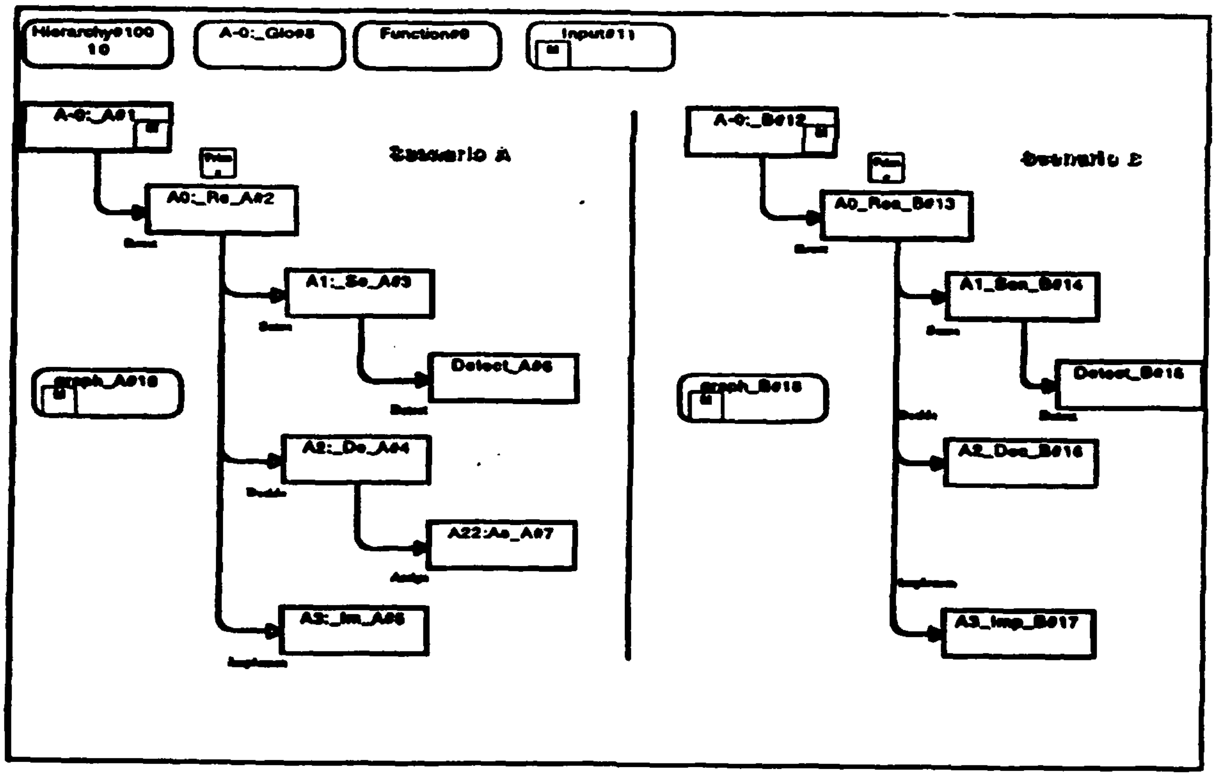
## DIAGRAMS OF THEATER MISSILE DEFENSE

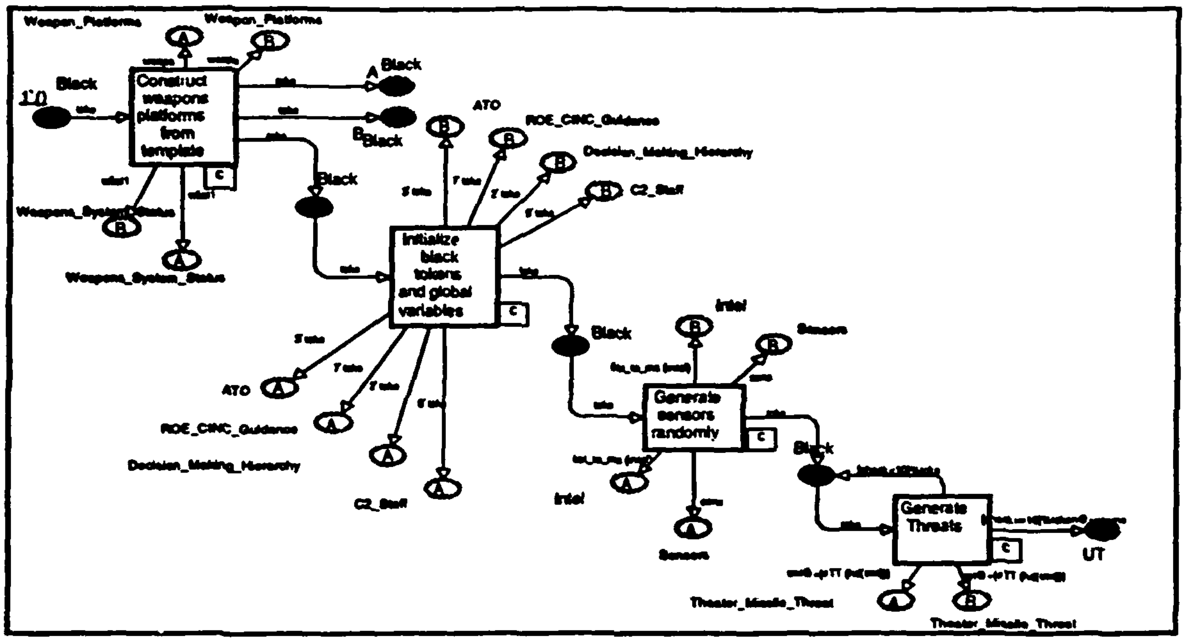
This is the "Hierarchy Page," the equivalent to the IDEF "node tree." Each node represents a page in the simulation. The Hierarchy node represents this page.

The two sides of the diagram represent the two "to-be" scenarios compared in the simulation. The difference between the two is in the absence of the A22 page. The A22 node does not need decomposition in Scenario B, and therefore this page does not exist.

The three nodes across the top contain the Global Declaration node, the Temporary Declaration node and the input functions. These contain code that applies to both scenarios. They contain the declarations of variables and functions necessary for the simulation as well as functions that generate input for both simulations to use.

Each scenario includes its own graphing setup page and corresponding node (graph_A and graph_B).






**Input Page:** The transitions on this page read from the input file and generate the necessary tokens to begin the simulation. Duplicate tokens are created so that Scenario A and Scenario B both operate with the same input data.

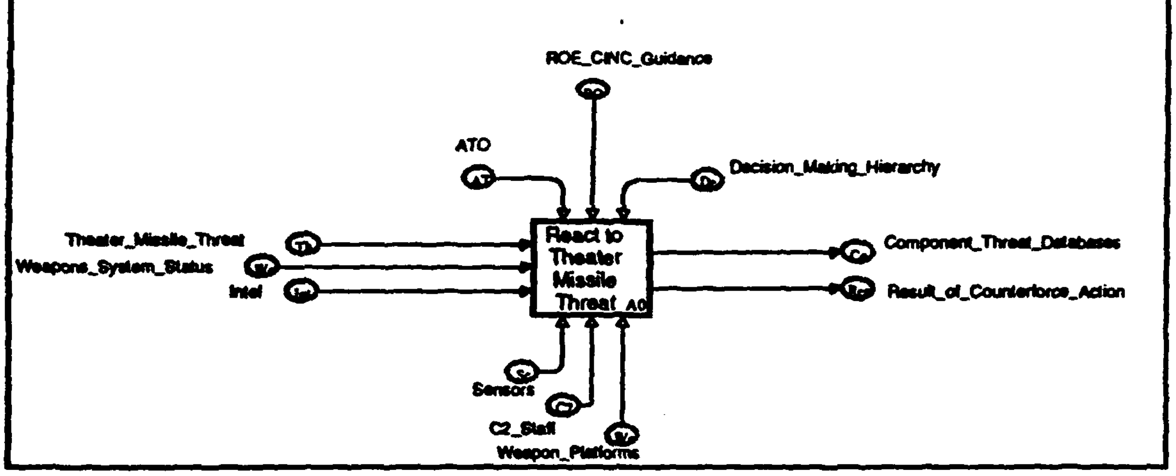
**Construct weapons platforms from template:** This transition generates the weapons platforms as well as a database with the status of each weapon.

**Initialize black tokens and global variables:** The black tokens represent inputs that are controls in IDEF0 and must be present for the activity to occur, e.g., rules of engagement. The attributes of these inputs are implemented through code and need not be passed with the tokens. The global variables are used to define decision criteria and can be changed in the input file.

**Generate sensors randomly** reads in the parameters describing the sensors and creates a list of sensors and their ranges, locations, and capabilities.

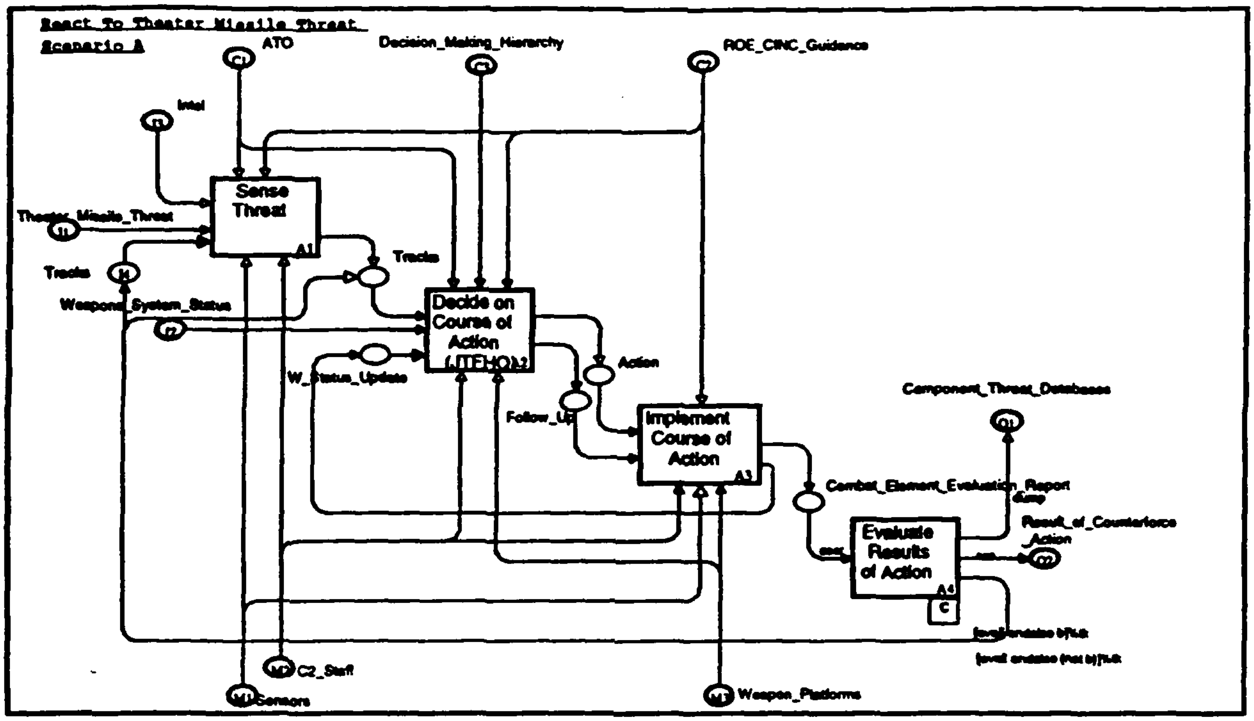
**Generate Threats** creates 10 sets of 1 to 4 threats at different times over a 3-day period. Once the threats are in place, the simulation can begin.

THEATER MISSILE DEFENSE/COUNTERFORCE
Scenario A




A-0 Scenario A

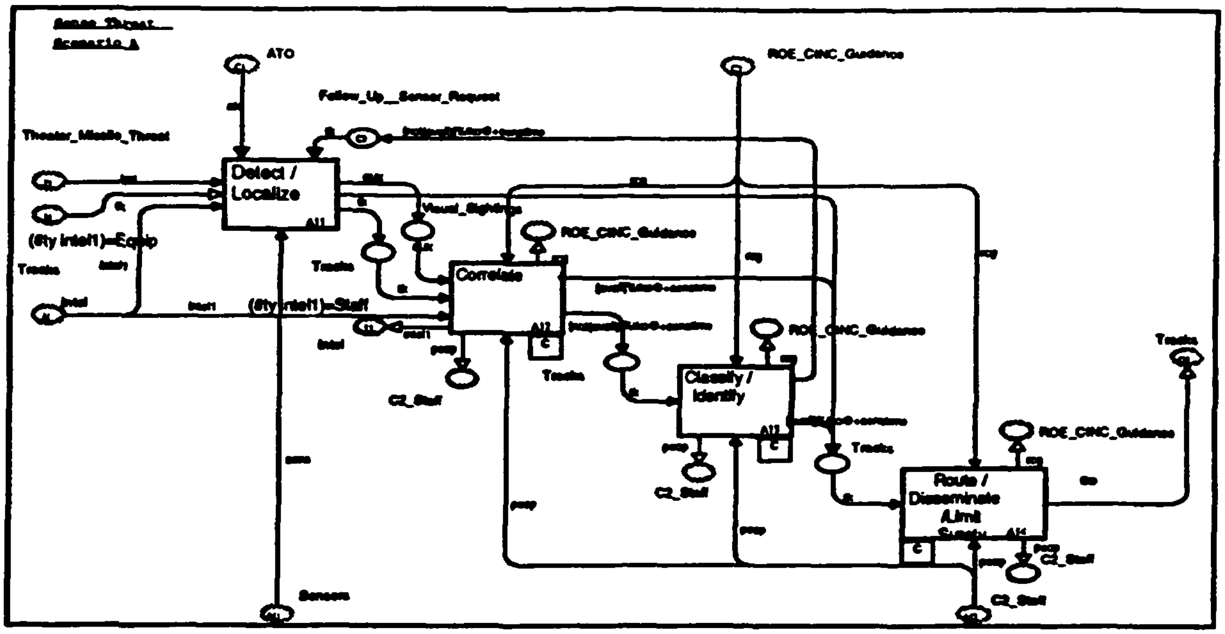
This page is the context page from the IDEF0 diagram. The rest of the Scenario A simulation is a decomposition of this box (transition). The places here are "fusion nodes"; they are fed by the transitions on the Input page described above. They feed into the decomposition page.




React to Theater Missile Threat (A0 node)
Scenario A

This is the first decomposition of the A-0 transition node. The transitions are the activities from IDEF0. The first three, **Sense Threat, Decide on Course of Action,** and **Implement Course of Action** are "substitution hierarchy transitions." This means they are further decomposed.

**Evaluate Results of Action:** Graphs are updated, and the decision is made whether to restrike the target. It has no further decomposition.

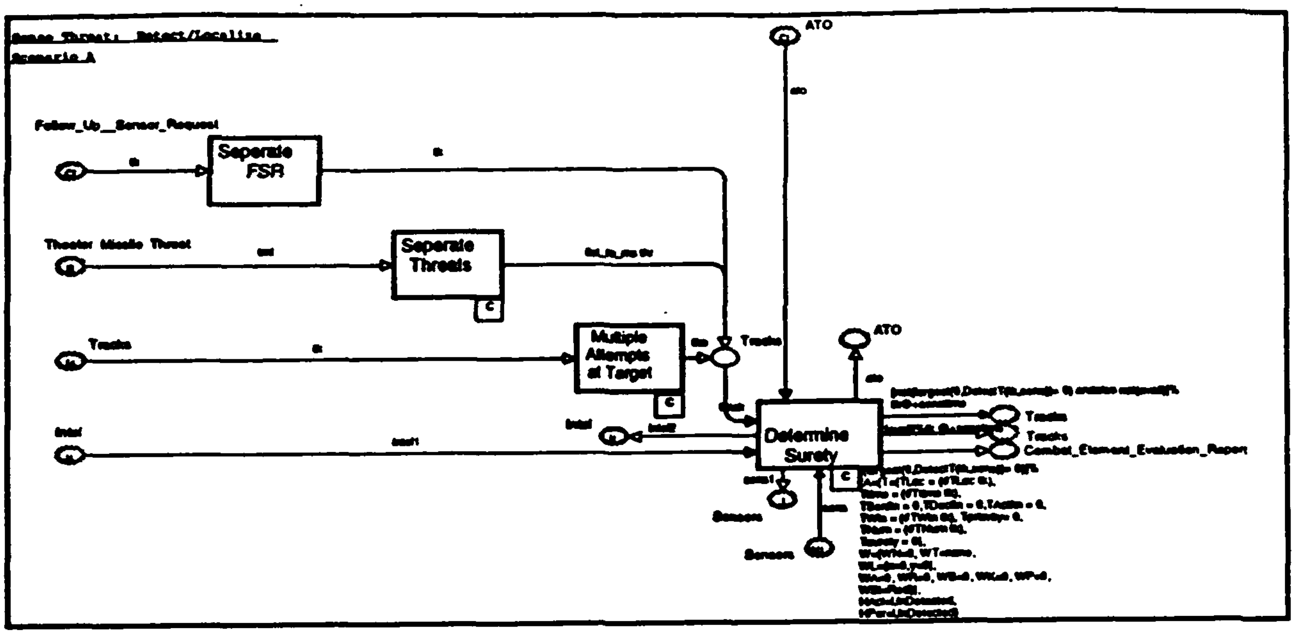



Sense Threat (A1)
Scenario A

**Detect/Localize** is not further decomposed in the IDEF0 model. However, it is decomposed in CPN to accommodate optional inputs. The transition should occur for the following input: (1) a new theater missile threat, (2) a followup sensor request (request for a second look at a target), or (3) a restrike at an already processed theater missile threat. CPN requires that all inputs be present for a transition to fire, so this must be resolved by adding separate transitions. This node is, therefore, a substitution hierarchy, and the issue is resolved at a lower level. Determination of whether or not the sensor array picked up the new threat occurs at this transition. A surety level is assigned within a range (a parameter in the input file) and used as a threshold of confidence in sensor activity.

**Correlate and Classify/Identify:** These two transitions represent intelligence work. Each adds surety based on the talent of the intelligence operator selected for the transition. At the end of each transition, time is checked. If too much time is elapsed, the threat is sent directly to be routed for action. Otherwise, it remains in the sensing/intelligence loop. A request for a second sensor look is generated if there is not enough surety and there is enough time remaining to do so. The surety threshold is a global variable established by the input files.

**Route/Disseminate/Limit Surety** sends those threats that the sensors have not detected directly to the evaluation transition. Threats that have been detected are passed on with the appropriate time stamp reflecting the amount of time spent in sensing the threat. Surety is limited to 100 percent.




Detect Localize (not in IDEF0)
Scenario A

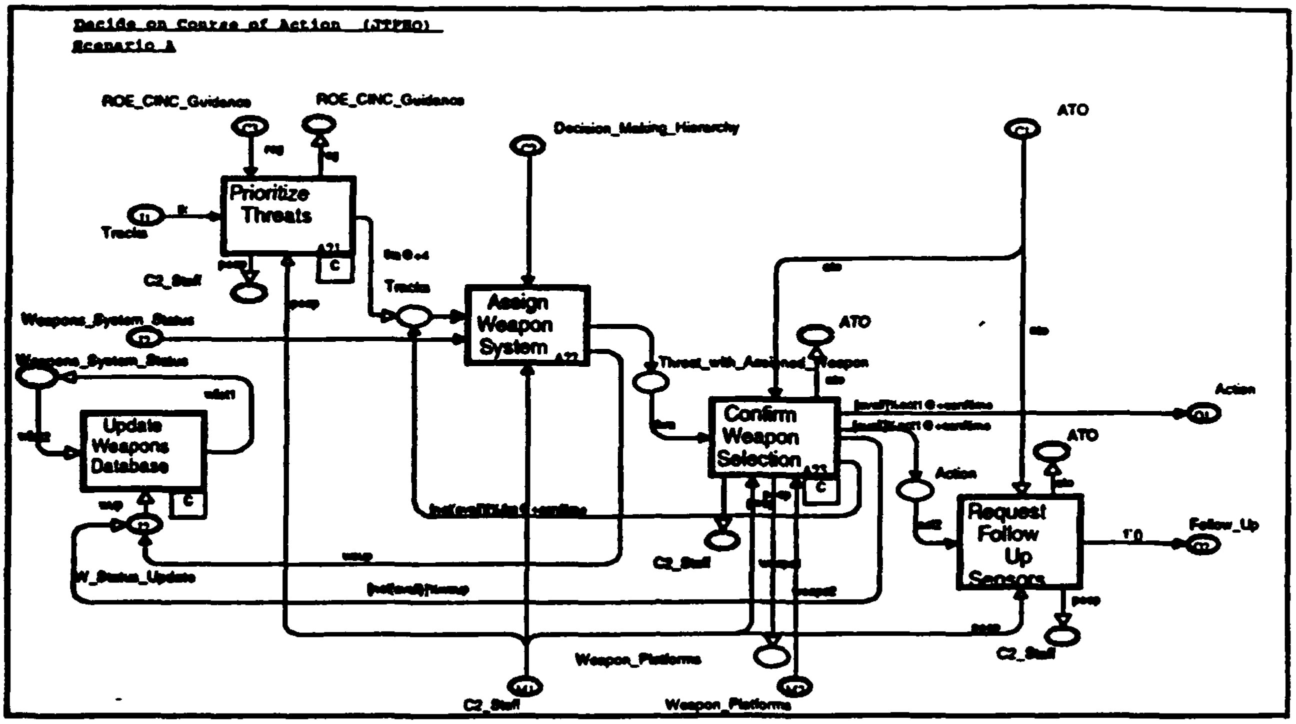
The optional inputs that presented a problem with the parent transition are handled by having a separate transition for each where the output is put into a standard format and feeds one place, which in turn feeds the transition that determines the surety.

**Separate FSR:** Follow-Up-Sensor-Requests are put into the necessary form for determining surety.

**Separate Threats:** The threat list is separated into individual threats for processing and tracking.

**Multiple Attempts at Target:** Tracks that have already had one or more strikes are prepared for another sensor look.

**Determine Surety:** Sensors and Intel operate on incoming threats (regardless of source), and a random draw determines the amount of surety added by the sensor look. It is determined which sensors have a good view of the threat, and the parameters of those sensors are applied.




Decide on Course of Action (A2)
Scenario A

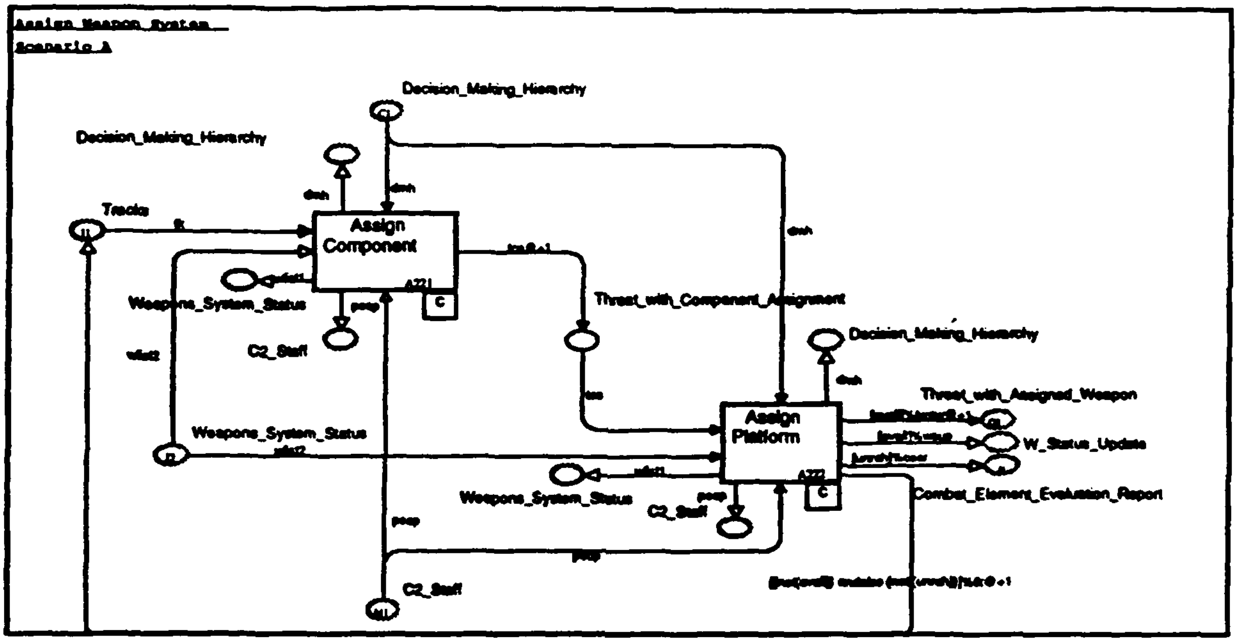
**Prioritize Threats** assigns a priority to each threat based on a matrix of surety and time since first sensing.

**Assign Weapon System** is where the difference between Scenario A and B occurs. In this case it is a substitution hierarchy transition. The decomposition will be described on the next page. The output is a threat with a weapon assignment. The assignment is made from a database that contains the last known status of each weapon.

**Confirm Weapon Selection** is intended to represent the occurrence of operational failure: an incidence of a red light on an aircraft panel or on a missile launcher. An update is made to the database if necessary.

**Request Followup Sensors** is intended to represent changes in tasking of sensors to evaluate strike result. Not implemented due to time constraints.

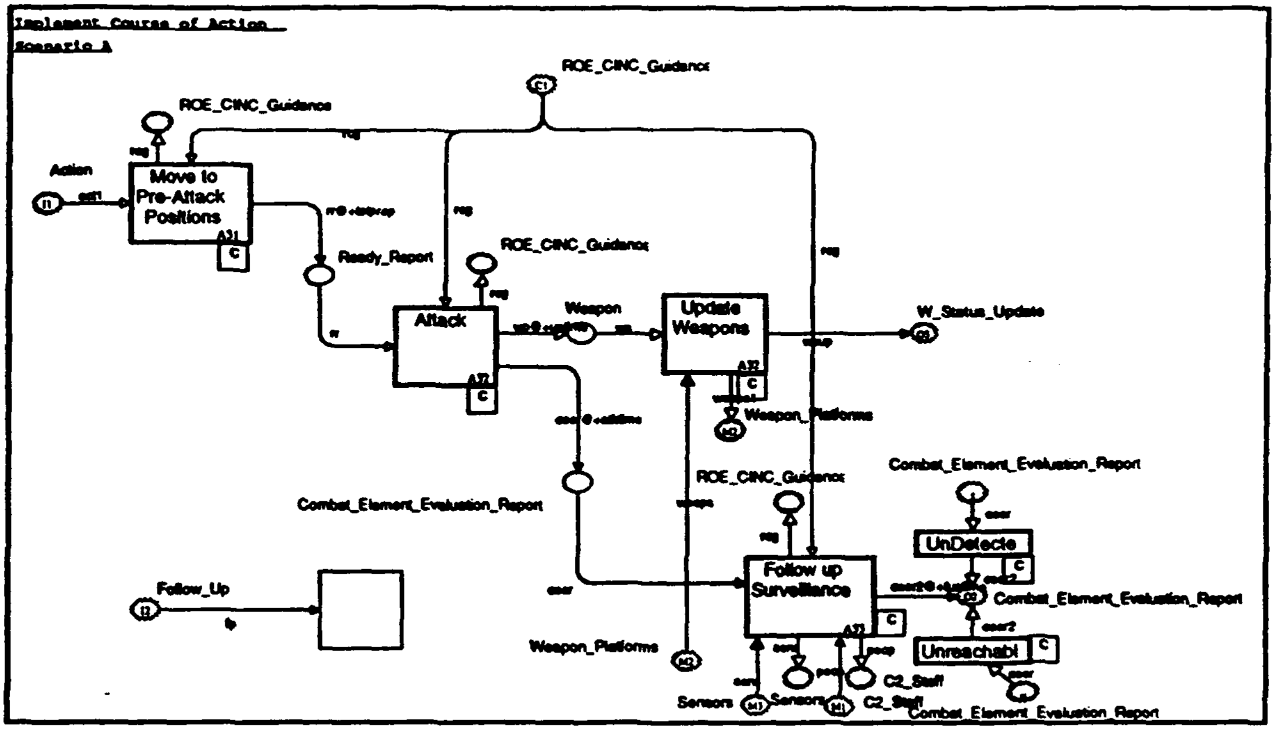
**Update Weapons Database** corresponds to an implied IDEF0 activity. The input Weapons-System-Status is assumed to be updated outside the IDEF0 model. It must be done explicitly in CPN. Thus, this transition receives changes in status and corrects the database to reflect the latest status.




Assign Weapon System (A22)
Scenario A

**Assign Component**: Represents Joint HQ assignment of a component to a threat based solely on geography.

**Assign Platform**: Transition in which the component commander assigns a specific weapon platform. The algorithm used is "fastest steel on target." The assignment is made from a database that contains the last known status of each weapon. If no weapon is available, the track is returned to Joint HQ for reassignment. If the threat is beyond the reach of available weapons and the window has closed, then it is passed on as "unreachable."




Implement Course of Action (A3)
Scenario A

**Move to Preattack Positions**, in the case of aircraft, includes the time necessary to move into the strike zone, which n the case of missiles includes only aiming time.

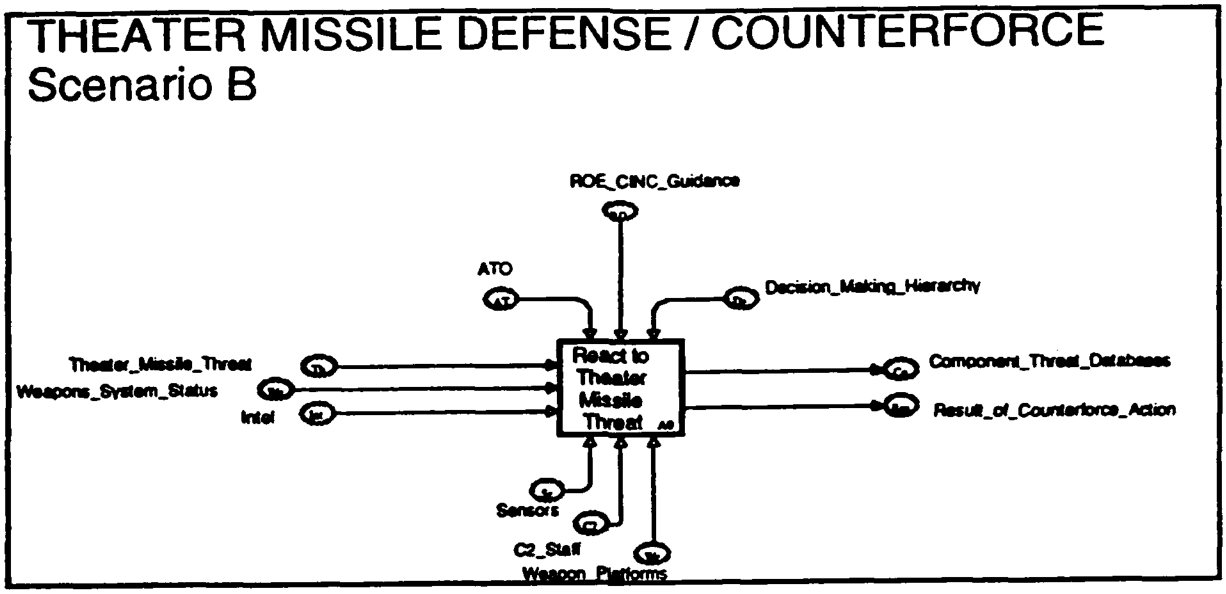
**Attack** is the transition in which random numbers are used to decide whether or not the target is hit and whether it is Killed or Maimed, given a hit.

**Update Weapons:** An update is sent to the weapons database. This takes place after the attack and enough time has elapsed for the weapon to be available again. In the case of aircraft, this is intended to include enough time for return to base and refueling and rearming activities.

**Followup Surveillance**: Transition in which the "perceived" result is determined.

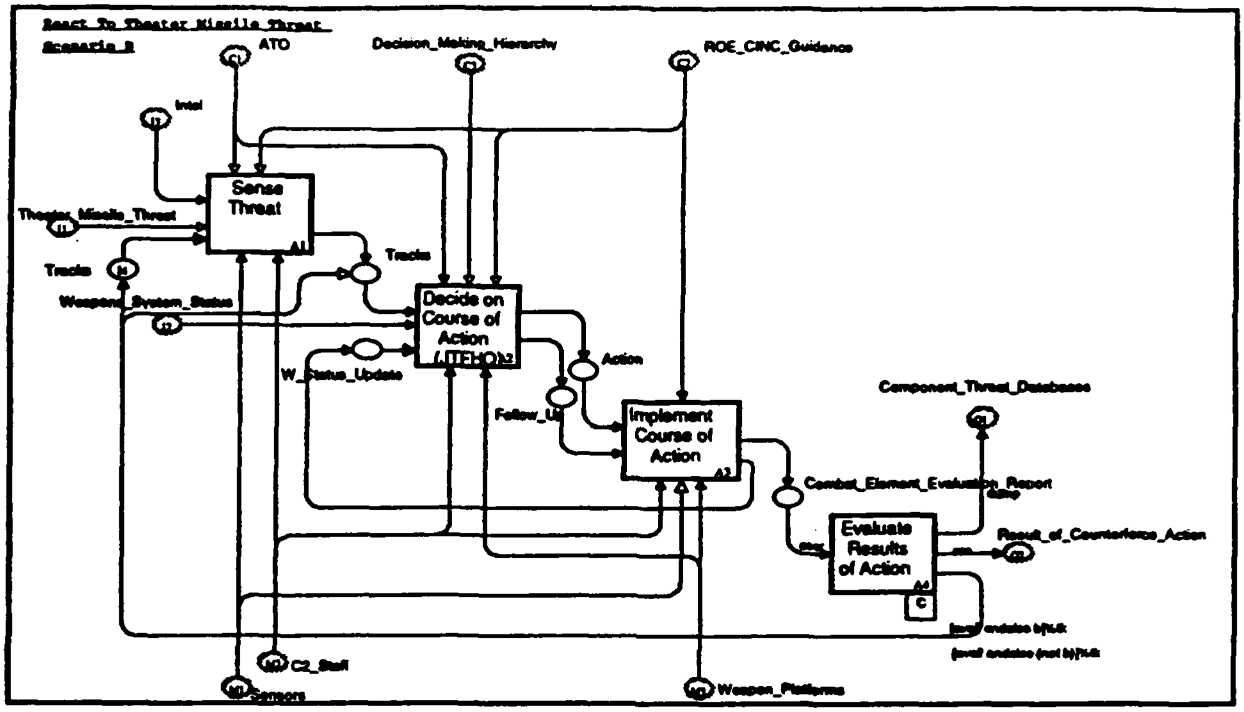
**Undetected**: Statistics are kept on undetected threats.

**Unreachable**: Statistics are kept on unreachable targets.




A-0 Scenario B

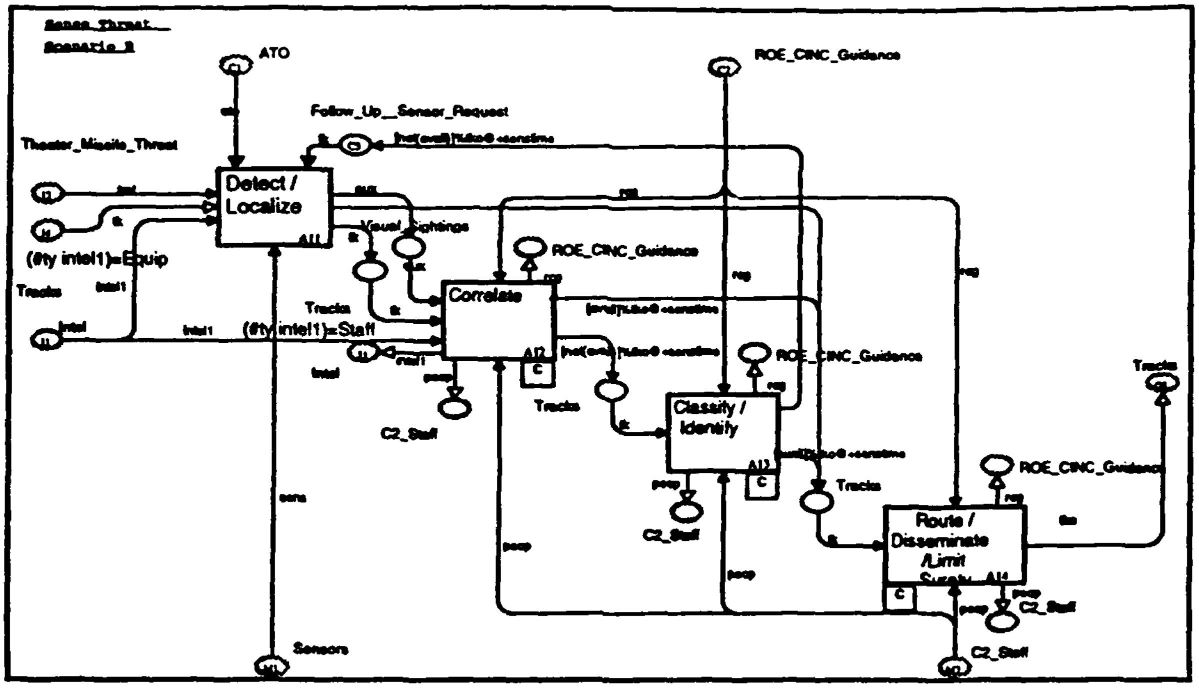
This page is the context page from the IDEF0 diagram. The rest of the Scenario B simulation is a decomposition of this box (transition). The places here are "fusion nodes"; they are fed by the transitions on the Input page described above. They feed into the decomposition page.




React to Theater Missile Threat (A0 node)
Scenario B

This is the first decomposition. The transitions are the activities from IDEF0. The first three, **Sense Threat, Decide on Course of Action,** and **Implement Course of Action** are "substitution hierarchy transitions." This means they are further decomposed.

**Evaluate Results of Action:** Graphs are updated, and the decision is made whether to restrike the target. It has no further decomposition.

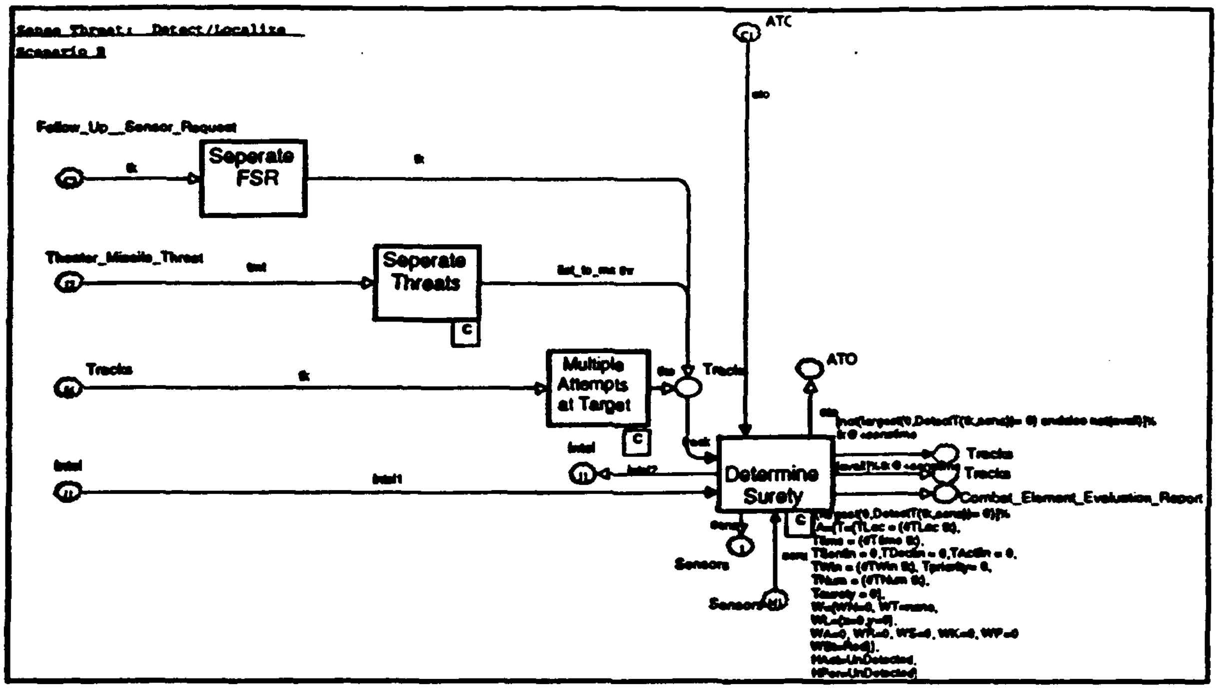



Sense Threat (A1)
Scenario B

**Detect/Localize** is not further decomposed in the IDEF0 model. However, it is decomposed in CPN to accommodate optional inputs. The transition should occur for the following input: (1) is a new theater missile threat, (2) a followup sensor request (request for a second look at a target), or (3) a restrike at an already processed theater missile threat. CPN requires that all inputs be present for a transition to fire, so this must be resolved by adding separate transitions. This is, therefore, a substitution hierarchy, and the issue is resolved at a lower level. Determination of whether or not the sensor array picked up the new threat occurs at this transition. A surety level is assigned within a range (a parameter in the input file) and used as a threshold of confidence in sensor activity.

**Correlate and Classify/Identify:** These two transitions represent intelligence work. Each adds surety based on the talent of the intelligence operator selected for the transition. At the end of each transition, time is checked. If too much time is elapsed, the threat is sent directly to be routed for action. Otherwise, it remains in the sensing/intelligence loop. A request for a second sensor look is generated if there is not enough surety and there is enough time remaining to do so. The surety threshold is a global variable established by the input files.

**Route/Disseminate/Limit Surety** sends those threats that the sensors have not detected directly to the evaluation transition. Threats that have been detected are passed on with the appropriate time stamp reflecting the amount of time spent in sensing the threat. Surety is limited to 100 percent.




Detect/Localize (not in IDEF0)
Scenario B

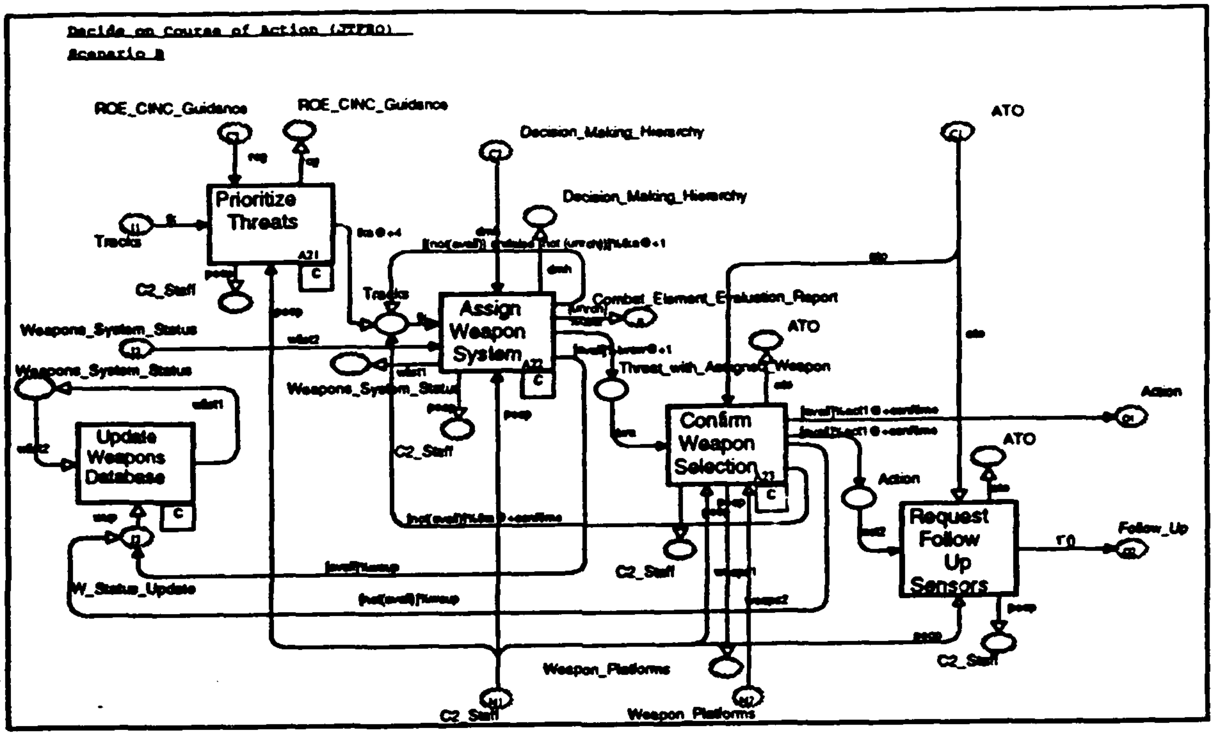
The optional inputs that presented a problem with the parent transition are handled by having a separate transition for each where the output is put into a standard format and feeds one place, which in turn feeds the transition that determines the surety.

**Separate FSR** is where Follow-Up-Sensor-Requests are put into the necessary form for determining surety.

**Separate Threats** is where the threat list is separated into individual threats for processing and tracking.

**Multiple Attempts at Target** is where tracks that have already had one or more strikes are prepared for another sensor look.

**Determine Surety:** Sensors and Intel operate on incoming threats (regardless of source) and a random draw determines the amount of surety added by the sensor look. It is determined which sensors have a good view of the threat, and the parameters of those sensors are applied.




Decide on Course of Action (A2)
Scenario B

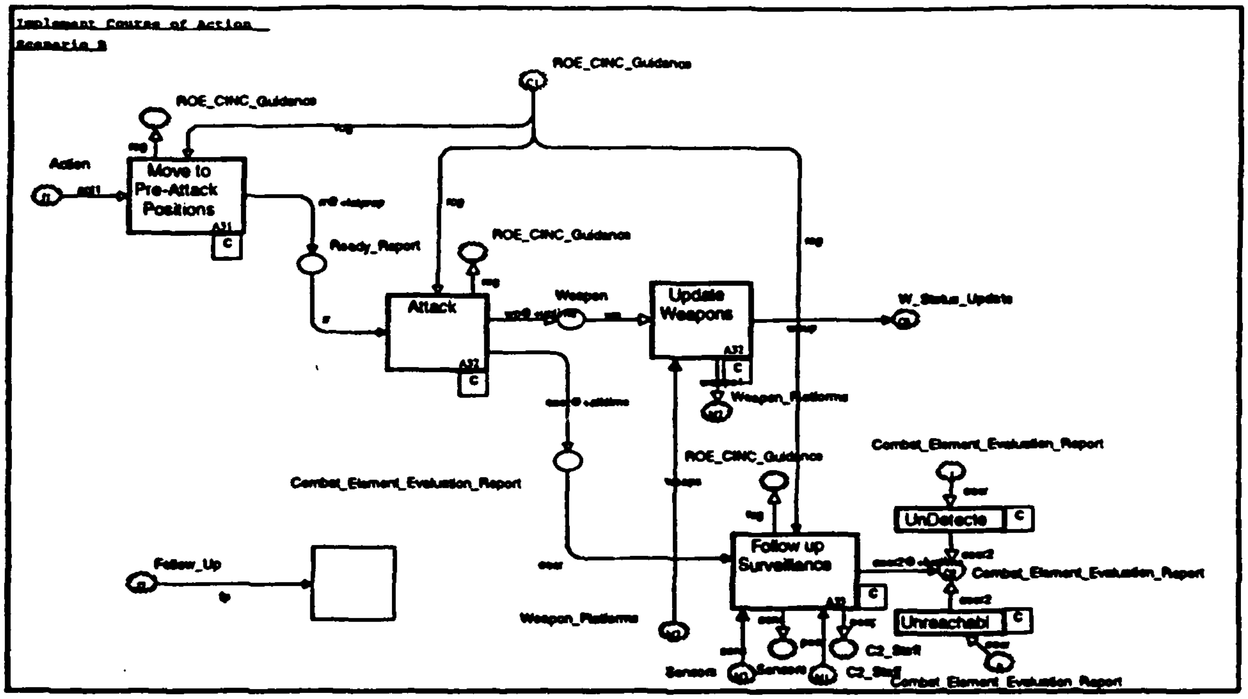
**Prioritize Threats** assigns a priority to each threat based on a matrix of surety and time since first sensing.

**Assign Weapon Svstem** is where the difference between Scenario A and B occurs. In this case it is ***not*** a su. stitution hierarchy transition. Instead, the weapon assignment is made in this transition to represent assets dedicated to the counterforce mission and given to the Joint Staff. The output is a threat with a weapon assignment. The assignment is made from a database that contains the last known status of each weapon.

**Confirm Weapon Selection** is intended to represent the occurrence of operational failure: an incidence of a red light on an aircraft panel or on a missile launcher. An update is made to the database if necessary.

**Request Followup Sensors** is intended to represent changes in tasking of sensors to evaluate strike result. Not implemented due to time constraints.

**Update Weapons Database:** This corresponds to an *implied* IDEF0 activity. The input Weapons-System-Status is assumed to be updated outside the IDEF0 model. It must be done explicitly in CPN. Thus, this transition receives changes in status and corrects the database to reflect the latest status.




Implement Course of Action (A3)
Scenario B

**Move to Preattack Positions**, in the case of aircraft, includes the time necessary to move into the strike zone, which in the case of missiles includes only aiming time.

**Attack** is the transition in which random numbers are used to decide whether or not the target is hit and whether it is Killed or Maimed, given a hit.

**Update Weapons** takes place after the attack and enough time has elapsed for the weapon to be available again. In the case of aircraft, this is intended to include enough time for return to base and refueling and rearming activities.

**Followup Surveillance**: Transition in which the "perceived" result is determined.

**Undetected**: Statistics are kept on undetected threats.

**Unreachable**: Statistics are kept on unreachable targets.

# APPENDIX D

# BIBLIOGRAPHY

# BIBLIOGRAPHY


James A. Brimson, Activity Accounting, An Activity Based Costing Approach, The Wiley/National Association of Accountants Professional Bank Series, John Wiley & Sons, Inc., New York, 1991.

Anne J. Clough, Charles Stark, Draper Laboratory, "Choosing an Appropriate Modeling Technology," Cross Talk, August 1992.

Colonel Gary Q. Coe, USA, "Measuring Force Multiplication," Proceedings of the 1988 Symposium on Command and Control Research, pp 63-69. Sponsored by Kazie Research Group Technical Panel on C3, Joint Directors of Laboratories and National Defense University and in cooperation with LEGG Control Systems Society, Sep 88.

Colonel Gary Q. Coe, USA, and Dr. John T. Dockery, "OJCS Initiatives in C2 Analysis and Simulation," Science of Command and Control, Coping with Uncertainty, pp 19-31, AFCEA International Press, Washington, DC, 1988.

Colonel Gary Q. Coe, USA, "The Search for Uncertainty in an Uncertain World" Professional Papers of AFCEA's 9th Western Conference, Anaheim, CA, 27 January 1988.

Department of Defense, DODI 8020.1 (Draft), "Functional Process Improvement Program," ASD (C3I), dated 1 October 1992.

Department of Defense Manual, DOD 8020.1M (Draft), "Functional Process Improvement (Functional Management Process for Implementing the Information Management Program of the Department of Defense)," Director of Defense Information, Office of the Secretary of Defense, August 1992.

Kathie Groveston, "Structured Analysis Model of Theater Missile Defense Counterforce Operations (U)," POET TMD Counterforce Study, 12 Feb 92.

GAO/IMTEC 92-3, Military Space Operations Satellite Control System Improved, But Serious Problems Remain (U), 27 Dec 91.

IDEF User's Group, User's Manual, ICAM Definition (IDEF) Method for Information Modeling (IDEF1X), IDEFUG-M-1X-8511, reproduced with the approval of WL/MTI (Air Force Wright Aeronautical Laboratories), Nov 1985.

IDEF User's Group, User's Manual, ICAM Definition Method for Function Modeling (IDEF0), IDEFUG-M-1X-8511, reproduced with the approval of WL/MTI (Air Force Wright Aeronautical Laboratories), Nov 1985.

Kurt Jensen, Soren Christiansen, Peter Huber, and Meers Holla, "Design CPN, A Reference Manual, Version 1.75," published by Meta Software, August 1991.

Meta Software Corp., "Design/CPN Tutorial Version 2.0 for the Macintosh," 2 April 1992.

J.L. Peterson, Petri Net Theory and the Modeling of Systems, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981.

Robin Ray, Janice Snow, and Barbara Dooley, User's Manual, Design/IDEF for the Apple Macintosh, published by Meta Software Corp, Cambridge, MA, Version 2.0, 1992.

SAF/AQS Staff Study, Air Force Satellite Control Network Capacity Management: An Analysis of Network Performance Management and Capacity Planning, HQ USAF, 19 Mar 92.

Paul Strassmann, The Business Value of Computers, An Executive's Guide, The Information Economics Press, New Canaan, CT, 1990.

Åke Wikström, Functional Programming Using Standard ML, Prentice Hall International Series in Computer Science, London, 1987.

# APPENDIX E

# DISTRIBUTION LIST

## DISTRIBUTION LIST

| Organization | Copies |
|---|---|
| Joint Interoperability & Engineering Organization (JIEO)<br>Lieutenant Colonel Rod Wijas, USMC | 10 |
| U.S. Space Command (USSPACECOM) | 3 |
| Strategic Defense Initiative Office (SDIO) | 3 |
| Director of Defense Information | 10 |
| Joint Chiefs of Staff | 2 |